Computer vision approaches to
Liquid-Phase Transmission Electron Microscopy

A dissertation presented for the degree of
Doctor of Philosophy

University College London, UK

Gabriele Marchello

Primary supervisor: Prof. Giuseppe Battaglia
Co-supervisor: Dr. Lorena Ruiz-Perez

March 2021
DECLARATION

I, Gabriele Marchello, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.
LIST OF PUBLICATIONS

Peer reviewed


Rebuttal


E-print on arXiv


To be submitted


In preparation


* These authors contributed equally
ABSTRACT

Electron microscopy (EM) is a technique that exploits the interaction between electron and matter to produce high resolution images down to atomic level. In order to avoid undesired scattering in the electron path, EM samples are conventionally imaged in solid state under vacuum conditions. Recently, this limit has been overcome by the realization of liquid-phase electron microscopy (LP EM), a technique that enables the analysis of samples in their liquid native state. LP EM paired with a high frame rate acquisition direct detection camera allows tracking the motion of particles in liquids, as well as their temporal dynamic processes.

In this research work, LP EM is adopted to image the dynamics of particles undergoing Brownian motion, exploiting their natural rotation to access all the particle views, in order to reconstruct their 3D structure via tomographic techniques. However, specific computer vision-based tools were designed around the limitations of LP EM in order to elaborate the results of the imaging process. Consequently, different deblurring and denoising approaches were adopted to improve the quality of the images. Therefore, the processed LP EM images were adopted to reconstruct the 3D model of the imaged samples. This task was performed by developing two different methods: Brownian tomography (BT) and Brownian particle analysis (BPA). The former tracks in time a single particle, capturing its dynamics evolution over time. The latter is an extension in time of the single particle analysis (SPA) technique. Conventionally it is paired to cryo-EM to reconstruct 3D density maps starting from thousands of EM images by capturing hundreds of particles of the same species frozen on a grid. On the contrary, BPA has the ability to process image sequences that may not contain thousands of particles, but instead monitors individual particle views across consecutive frames, rather than across a single frame.
The work presented in this thesis investigates the field of liquid-phase electron microscopy (LP EM), a technique enabling to image samples in their liquid native state. Unfortunately, LP EM is a relatively recent technique, most of aspects of which are still unknown. In this research work, LP EM technique is widely investigated, mainly focusing on the recorded images. Consequently, some of the limitations characterising LP EM were identified, and several computer vision methods were proposed in order to push further and further such limits.

The majority of the results achieved in this research work was presented at different conferences. In particular, the proposed 3D reconstruction method was presented with a poster at the NanoSpainConf in 2019 and at the 17th IBEC Symposium, while with an oral presentation at NEUBIAS 2019. Additionally, the 3D reconstruction method paired with the dynamics simulation was initially submitted to Nature, but was not published. However, the comments received by the reviewers were used to improve the quality and the robustness of the results presented in the manuscript. At the time the thesis is being written, the latest version of the manuscript is being prepared for a second submission. Moreover, other methods, such as the image denoising pipeline, and the method to identify particles in dry state have been published in Journal of Microscopy, and Nature Communication, respectively. The methodology designed for both works was upload in the GitHub folder of the author and forked into the folder of the Molecular Bionics research group. In this fashion, the results accomplished in this thesis may be not only shared with, but also exploited by other researchers.

On a personal side, working on a cutting-edge technique as LP EM represented a great opportunity to step out of the comfort zone, facing a field with scarce knowledge. Moreover, it enabled a deep dive into the computer vision field, exploring the capabilities of artificial intelligence – and in particular learning-based methods – when applied to images.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 INTRODUCTION</td>
<td>15</td>
</tr>
<tr>
<td>1.1 Thesis organisation</td>
<td>18</td>
</tr>
<tr>
<td>2 LITERATURE REVIEW</td>
<td>19</td>
</tr>
<tr>
<td>2.1 Electron Microscopy</td>
<td>19</td>
</tr>
<tr>
<td>2.1.1 The electron microscope</td>
<td>19</td>
</tr>
<tr>
<td>2.1.2 Transmission electron microscope</td>
<td>27</td>
</tr>
<tr>
<td>2.1.3 Scanning electron microscope</td>
<td>27</td>
</tr>
<tr>
<td>2.1.4 Scanning transmission electron microscope</td>
<td>28</td>
</tr>
<tr>
<td>2.2 Sample Preparation</td>
<td>28</td>
</tr>
<tr>
<td>2.2.1 Staining</td>
<td>29</td>
</tr>
<tr>
<td>2.2.2 Cryo-EM</td>
<td>30</td>
</tr>
<tr>
<td>2.3 3D Reconstruction</td>
<td>32</td>
</tr>
<tr>
<td>2.3.1 Electron tomography</td>
<td>33</td>
</tr>
<tr>
<td>2.3.2 Single particle analysis</td>
<td>35</td>
</tr>
<tr>
<td>2.3.3 Resolution</td>
<td>41</td>
</tr>
<tr>
<td>2.4 Liquid-Phase Electron Microscopy</td>
<td>43</td>
</tr>
<tr>
<td>2.4.1 Brownian motion</td>
<td>47</td>
</tr>
<tr>
<td>2.4.2 Nanoparticle imaging</td>
<td>48</td>
</tr>
<tr>
<td>2.5 Computer Vision</td>
<td>50</td>
</tr>
<tr>
<td>2.5.1 Traditional programming versus Machine Learning</td>
<td>51</td>
</tr>
<tr>
<td>2.5.2 Machine Learning</td>
<td>52</td>
</tr>
<tr>
<td>2.5.2.1 Human supervision classification criterion</td>
<td>53</td>
</tr>
<tr>
<td>2.5.2.2 Generalisation classification criterion</td>
<td>53</td>
</tr>
<tr>
<td>2.5.2.3 Learning classification criterion</td>
<td>54</td>
</tr>
<tr>
<td>2.5.3 Artificial neural network</td>
<td>55</td>
</tr>
<tr>
<td>2.5.4 Convolutional neural network</td>
<td>59</td>
</tr>
<tr>
<td>2.6 Computer Vision Applied to Electron Microscopy</td>
<td>61</td>
</tr>
<tr>
<td>2.6.1 Image enhancement</td>
<td>61</td>
</tr>
<tr>
<td>2.6.2 Particle detection</td>
<td>64</td>
</tr>
<tr>
<td>3 METHODOLOGY</td>
<td>66</td>
</tr>
<tr>
<td>3.1 The Equipment</td>
<td>66</td>
</tr>
<tr>
<td>3.2 Noise Reduction</td>
<td>70</td>
</tr>
<tr>
<td>3.2.1 End-to-end pipeline</td>
<td>71</td>
</tr>
<tr>
<td>3.2.2 Denoising autoencoder</td>
<td>75</td>
</tr>
<tr>
<td>3.2.3 Noise2Void</td>
<td>77</td>
</tr>
<tr>
<td>3.3 Particles Identification</td>
<td>80</td>
</tr>
<tr>
<td>3.3.1 Identification algorithm v1.0</td>
<td>81</td>
</tr>
<tr>
<td>3.3.2 Identification algorithm v2.0</td>
<td>82</td>
</tr>
<tr>
<td>3.3.3 Deep LiqID</td>
<td>84</td>
</tr>
<tr>
<td>3.3.4 Identification in dry state</td>
<td>89</td>
</tr>
<tr>
<td>3.4 Tracking Algorithm</td>
<td>91</td>
</tr>
<tr>
<td>3.5 3D Reconstruction</td>
<td>94</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 RESULTS AND DISCUSSION</td>
<td>98</td>
</tr>
<tr>
<td>4.1 Fluid Dynamic Simulation</td>
<td>98</td>
</tr>
<tr>
<td>4.2 Imaging Analysis</td>
<td>100</td>
</tr>
<tr>
<td>4.3 Noise Reduction</td>
<td>104</td>
</tr>
<tr>
<td>4.3.1 End-to-end pipeline</td>
<td>105</td>
</tr>
<tr>
<td>4.3.2 Denoising autoencoder</td>
<td>110</td>
</tr>
<tr>
<td>4.3.3 Noise2Void</td>
<td>113</td>
</tr>
<tr>
<td>4.4 Particle Identification</td>
<td>118</td>
</tr>
<tr>
<td>4.4.1 Traditional algorithm v1.0</td>
<td>119</td>
</tr>
<tr>
<td>4.4.2 Traditional algorithm v2.0</td>
<td>124</td>
</tr>
<tr>
<td>4.4.3 Deep LiqID</td>
<td>126</td>
</tr>
<tr>
<td>4.4.4 Comparison of identification methods</td>
<td>131</td>
</tr>
<tr>
<td>4.4.5 Identification in dry state</td>
<td>133</td>
</tr>
<tr>
<td>4.5 Particles motion tracking</td>
<td>135</td>
</tr>
<tr>
<td>4.6 3D reconstruction</td>
<td>139</td>
</tr>
<tr>
<td>5 CONCLUSIONS</td>
<td>164</td>
</tr>
<tr>
<td>APPENDIX</td>
<td>168</td>
</tr>
<tr>
<td>CITED LITERATURE</td>
<td>171</td>
</tr>
</tbody>
</table>
## LIST OF TABLES

<table>
<thead>
<tr>
<th>TABLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>88</td>
</tr>
<tr>
<td>II</td>
<td>102</td>
</tr>
<tr>
<td>III</td>
<td>105</td>
</tr>
<tr>
<td>IV</td>
<td>106</td>
</tr>
<tr>
<td>V</td>
<td>113</td>
</tr>
<tr>
<td>VI</td>
<td>114</td>
</tr>
<tr>
<td>VII</td>
<td>117</td>
</tr>
<tr>
<td>VIII</td>
<td>118</td>
</tr>
<tr>
<td>IX</td>
<td>120</td>
</tr>
<tr>
<td>X</td>
<td>122</td>
</tr>
<tr>
<td>XI</td>
<td>122</td>
</tr>
<tr>
<td>XII</td>
<td>129</td>
</tr>
<tr>
<td>XIII</td>
<td>132</td>
</tr>
<tr>
<td>XIV</td>
<td>142</td>
</tr>
<tr>
<td>XV</td>
<td>144</td>
</tr>
<tr>
<td>XVI</td>
<td>144</td>
</tr>
<tr>
<td>XVII</td>
<td>146</td>
</tr>
<tr>
<td>XVIII</td>
<td>147</td>
</tr>
<tr>
<td>XIX</td>
<td>148</td>
</tr>
<tr>
<td>XX</td>
<td>149</td>
</tr>
<tr>
<td>XXI</td>
<td>151</td>
</tr>
<tr>
<td>XXII</td>
<td>153</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>20</td>
</tr>
<tr>
<td>2.2</td>
<td>21</td>
</tr>
<tr>
<td>2.3</td>
<td>22</td>
</tr>
<tr>
<td>2.4</td>
<td>23</td>
</tr>
<tr>
<td>2.5</td>
<td>24</td>
</tr>
<tr>
<td>2.6</td>
<td>26</td>
</tr>
<tr>
<td>2.7</td>
<td>28</td>
</tr>
<tr>
<td>2.8</td>
<td>31</td>
</tr>
<tr>
<td>2.9</td>
<td>31</td>
</tr>
<tr>
<td>2.10</td>
<td>33</td>
</tr>
<tr>
<td>2.11</td>
<td>34</td>
</tr>
<tr>
<td>2.12</td>
<td>35</td>
</tr>
<tr>
<td>2.13</td>
<td>36</td>
</tr>
<tr>
<td>2.14</td>
<td>36</td>
</tr>
<tr>
<td>2.15</td>
<td>39</td>
</tr>
<tr>
<td>2.16</td>
<td>40</td>
</tr>
<tr>
<td>2.17</td>
<td>40</td>
</tr>
<tr>
<td>2.18</td>
<td>42</td>
</tr>
<tr>
<td>2.19</td>
<td>44</td>
</tr>
<tr>
<td>2.20</td>
<td>46</td>
</tr>
<tr>
<td>2.21</td>
<td>46</td>
</tr>
<tr>
<td>2.22</td>
<td>48</td>
</tr>
<tr>
<td>2.23</td>
<td>49</td>
</tr>
<tr>
<td>2.24</td>
<td>51</td>
</tr>
<tr>
<td>2.25</td>
<td>52</td>
</tr>
<tr>
<td>2.26</td>
<td>55</td>
</tr>
<tr>
<td>2.27</td>
<td>56</td>
</tr>
<tr>
<td>2.28</td>
<td>57</td>
</tr>
<tr>
<td>2.29</td>
<td>58</td>
</tr>
<tr>
<td>2.30</td>
<td>60</td>
</tr>
<tr>
<td>2.31</td>
<td>61</td>
</tr>
<tr>
<td>2.32</td>
<td>62</td>
</tr>
<tr>
<td>2.33</td>
<td>63</td>
</tr>
<tr>
<td>3.1</td>
<td>66</td>
</tr>
<tr>
<td>3.2</td>
<td>68</td>
</tr>
<tr>
<td>3.3</td>
<td>69</td>
</tr>
<tr>
<td>3.4</td>
<td>72</td>
</tr>
<tr>
<td>3.5</td>
<td>77</td>
</tr>
<tr>
<td>3.6</td>
<td>78</td>
</tr>
<tr>
<td>3.7</td>
<td>80</td>
</tr>
<tr>
<td>3.8</td>
<td>82</td>
</tr>
<tr>
<td>3.9</td>
<td>83</td>
</tr>
<tr>
<td>3.10</td>
<td>83</td>
</tr>
<tr>
<td>3.11</td>
<td>84</td>
</tr>
<tr>
<td>FIGURE</td>
<td>PAGE</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>3.12</td>
<td>YOLO working principle</td>
</tr>
<tr>
<td>3.13</td>
<td>Anchor boxes and IoU</td>
</tr>
<tr>
<td>3.14</td>
<td>Deep LiqID database</td>
</tr>
<tr>
<td>3.15</td>
<td>Particle membranes measurement</td>
</tr>
<tr>
<td>3.16</td>
<td>Identification of particles in dry state algorithm</td>
</tr>
<tr>
<td>3.17</td>
<td>Particles tracking in time</td>
</tr>
<tr>
<td>3.18</td>
<td>Particle identification process via EMAN2 in Scipion</td>
</tr>
<tr>
<td>3.19</td>
<td>Flowchart of the reconstruction workflow in Scipion</td>
</tr>
<tr>
<td>4.1</td>
<td>Top view of the simulated holder</td>
</tr>
<tr>
<td>4.2</td>
<td>Results of the fluid dynamics simulation of the liquid holder</td>
</tr>
<tr>
<td>4.3</td>
<td>Frame rate analysis</td>
</tr>
<tr>
<td>4.4</td>
<td>PID algorithm for frame rate analysis</td>
</tr>
<tr>
<td>4.5</td>
<td>PSNR videos estimation</td>
</tr>
<tr>
<td>4.6</td>
<td>PSNR videos estimation</td>
</tr>
<tr>
<td>4.7</td>
<td>End-to-end denoising pipeline iterations</td>
</tr>
<tr>
<td>4.8</td>
<td>Comparison of deblurring algorithms</td>
</tr>
<tr>
<td>4.9</td>
<td>Performance analysis of the LED method</td>
</tr>
<tr>
<td>4.10</td>
<td>Pipeline intermediate results</td>
</tr>
<tr>
<td>4.11</td>
<td>Membrane analysis</td>
</tr>
<tr>
<td>4.12</td>
<td>DAE test images</td>
</tr>
<tr>
<td>4.13</td>
<td>DAE loss results</td>
</tr>
<tr>
<td>4.14</td>
<td>DAE loss results</td>
</tr>
<tr>
<td>4.15</td>
<td>DAE implementations results</td>
</tr>
<tr>
<td>4.16</td>
<td>N2V low dose input images</td>
</tr>
<tr>
<td>4.17</td>
<td>N2V low dose output images</td>
</tr>
<tr>
<td>4.18</td>
<td>N2V low dose details comparison</td>
</tr>
<tr>
<td>4.19</td>
<td>N2V training set for high quality results</td>
</tr>
<tr>
<td>4.20</td>
<td>N2V high quality results</td>
</tr>
<tr>
<td>4.21</td>
<td>N2V high quality results on STEM images</td>
</tr>
<tr>
<td>4.22</td>
<td>Processing stages of the traditional identification algorithm v1.0</td>
</tr>
<tr>
<td>4.23</td>
<td>Results of the traditional identification algorithm v1.0</td>
</tr>
<tr>
<td>4.24</td>
<td>Results of the traditional identification algorithm v1.0 in noisy images</td>
</tr>
<tr>
<td>4.25</td>
<td>Identification algorithm v1.0 applied to consecutive frames</td>
</tr>
<tr>
<td>4.26</td>
<td>Identification algorithm v2.0 local and global threshold</td>
</tr>
<tr>
<td>4.27</td>
<td>Identification algorithm v2.0 results</td>
</tr>
<tr>
<td>4.28</td>
<td>Identification algorithm v2.0 applied to consecutive frames</td>
</tr>
<tr>
<td>4.29</td>
<td>Training results of Deep LiqID over 680 for 1000 epochs</td>
</tr>
<tr>
<td>4.30</td>
<td>Training results of Deep LiqID over 780 for 1200 epochs</td>
</tr>
<tr>
<td>4.31</td>
<td>Training results of Deep LiqID over 780 for 2000 epochs</td>
</tr>
<tr>
<td>4.32</td>
<td>Training results of Deep LiqID over 780 for 1000 epochs</td>
</tr>
<tr>
<td>4.33</td>
<td>DeepLiqID confidence scores</td>
</tr>
<tr>
<td>4.34</td>
<td>Comparison of identification methods</td>
</tr>
<tr>
<td>4.35</td>
<td>Results of the identification in dry-state method</td>
</tr>
<tr>
<td>4.36</td>
<td>Particle trajectories reconstruction in image sequences with low SNR</td>
</tr>
<tr>
<td>4.37</td>
<td>Particle trajectories reconstruction in image sequences with high SNR</td>
</tr>
<tr>
<td>FIGURE</td>
<td>PAGE</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>4.39</td>
<td>MSD values for low and high SNR image sequences</td>
</tr>
<tr>
<td>4.38</td>
<td>Highlight of low resolution</td>
</tr>
<tr>
<td>4.40</td>
<td>Disk-like micelle of PMPC-PDPA</td>
</tr>
<tr>
<td>4.41</td>
<td>3D reconstruction of the disk-like micelle</td>
</tr>
<tr>
<td>4.42</td>
<td>Disk-like micelle of PMPC-PDPA profiles evolution</td>
</tr>
<tr>
<td>4.43</td>
<td>Validation of the PMPC-PDPA reconstructed model</td>
</tr>
<tr>
<td>4.44</td>
<td>Image sequence of protein ferritin for BPA</td>
</tr>
<tr>
<td>4.45</td>
<td>Protein ferritin BPA reconstruction</td>
</tr>
<tr>
<td>4.46</td>
<td>Comparison of the ferritin model obtained via BPA and x-ray tomography</td>
</tr>
<tr>
<td>4.47</td>
<td>Protein ferritin BPA reconstructions in time</td>
</tr>
<tr>
<td>4.48</td>
<td>Resolution distribution of the ferritin models in time</td>
</tr>
<tr>
<td>4.49</td>
<td>Ferritin protein imaged in low dose</td>
</tr>
<tr>
<td>4.50</td>
<td>Averaged frames of low dose ferritin image sequence</td>
</tr>
<tr>
<td>4.51</td>
<td>Ferritin reconstructed via BPA at low dose</td>
</tr>
<tr>
<td>4.52</td>
<td>Validation of the BPA at low dose</td>
</tr>
<tr>
<td>4.53</td>
<td>Comparison between x-ray and cryo-EM density maps</td>
</tr>
<tr>
<td>4.54</td>
<td>Ferritin structure deriving from the superimposition of 60 different conformations from molecular dynamics simulation</td>
</tr>
<tr>
<td>4.55</td>
<td>Molecular dynamics simulation of the structure of ferritin protein over time.</td>
</tr>
<tr>
<td>4.56</td>
<td>Molecular dynamics simulation of the unit of ferritin protein</td>
</tr>
<tr>
<td>4.57</td>
<td>Ferritin protein image sequences used to perform BPA and BT analyses</td>
</tr>
<tr>
<td>4.58</td>
<td>Comparison of the performance of EMAN2 and Deep LiqID</td>
</tr>
<tr>
<td>4.59</td>
<td>Ferritin protein 3D models reconstructed via BPA applied to high and low dose image sequences processed via N2V.</td>
</tr>
<tr>
<td>4.60</td>
<td>Time analysis of ferritin protein 3D models reconstructed via BPA applied to high dose image sequences processed via N2V.</td>
</tr>
<tr>
<td>4.61</td>
<td>Ferritin protein 3D models reconstructed via BPA applied to low dose image sequences processed via N2V.</td>
</tr>
<tr>
<td>4.62</td>
<td>BT analysis performed on high dose image sequence.</td>
</tr>
<tr>
<td>4.63</td>
<td>BT analysis performed on low dose image sequence.</td>
</tr>
<tr>
<td>4.64</td>
<td>BT and BPA obtained refining dynamical maps</td>
</tr>
<tr>
<td>4.65</td>
<td>Evaluation of the importance of initial model</td>
</tr>
</tbody>
</table>
## LIST OF ABBREVIATIONS

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1D</td>
<td>One-dimensional</td>
</tr>
<tr>
<td>2D</td>
<td>Two-dimensional</td>
</tr>
<tr>
<td>3D</td>
<td>Three-dimensional</td>
</tr>
<tr>
<td>4D</td>
<td>Four-dimensional</td>
</tr>
<tr>
<td>ADC</td>
<td>Analog-to-digital converter</td>
</tr>
<tr>
<td>ADF</td>
<td>Annular dark field</td>
</tr>
<tr>
<td>AI</td>
<td>Artificial intelligence</td>
</tr>
<tr>
<td>ANN</td>
<td>Artificial neural network</td>
</tr>
<tr>
<td>b</td>
<td>Bias term</td>
</tr>
<tr>
<td>B</td>
<td>Magnetic field</td>
</tr>
<tr>
<td>BCE</td>
<td>Binary cross-entropy</td>
</tr>
<tr>
<td>BPA</td>
<td>Brownian particle analysis</td>
</tr>
<tr>
<td>BT</td>
<td>Brownian tomography</td>
</tr>
<tr>
<td>c(·,·)</td>
<td>SSIM contrast term</td>
</tr>
<tr>
<td>Ci</td>
<td>SSIM constants</td>
</tr>
<tr>
<td>CCD</td>
<td>Charge-coupled device</td>
</tr>
<tr>
<td>CMOS</td>
<td>Complementary Metal Oxide Semiconductor</td>
</tr>
<tr>
<td>CNN</td>
<td>Convolutional neural network</td>
</tr>
<tr>
<td>ConvNet</td>
<td>Convolutional neural network</td>
</tr>
<tr>
<td>CPU</td>
<td>Central processing unit</td>
</tr>
<tr>
<td>cryo-EM</td>
<td>Cryogenic electron microscopy</td>
</tr>
<tr>
<td>CTF</td>
<td>Contrast transfer function</td>
</tr>
<tr>
<td>CV</td>
<td>Computer vision</td>
</tr>
<tr>
<td>d</td>
<td>Distance</td>
</tr>
<tr>
<td>D</td>
<td>Diameter</td>
</tr>
</tbody>
</table>
### LIST OF ABBREVIATIONS

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dB</td>
<td>Decibel</td>
</tr>
<tr>
<td>DAE</td>
<td>Denoising autoencoder</td>
</tr>
<tr>
<td>DDD</td>
<td>Direct detection device</td>
</tr>
<tr>
<td>DDID</td>
<td>Dual-domain image denoising</td>
</tr>
<tr>
<td>DL</td>
<td>Deep learning</td>
</tr>
<tr>
<td>$d_m$</td>
<td>Membrane thickness</td>
</tr>
<tr>
<td>DPR</td>
<td>Differential Phase Residual</td>
</tr>
<tr>
<td>DQE</td>
<td>Detection quantum efficiency</td>
</tr>
<tr>
<td>$D_r$</td>
<td>Rotational diffusion coefficient</td>
</tr>
<tr>
<td>$D_t$</td>
<td>Translational diffusion coefficient</td>
</tr>
<tr>
<td>EM</td>
<td>Electron microscopy</td>
</tr>
<tr>
<td>ETEM</td>
<td>Environmental Transmission Electron Microscopy</td>
</tr>
<tr>
<td>f</td>
<td>Activation function</td>
</tr>
<tr>
<td>$F_C$</td>
<td>Coulomb force</td>
</tr>
<tr>
<td>FC</td>
<td>Fully connected layer</td>
</tr>
<tr>
<td>FEG</td>
<td>Field emission gun</td>
</tr>
<tr>
<td>$F_L$</td>
<td>Lorentz force</td>
</tr>
<tr>
<td>fps</td>
<td>Frames per second</td>
</tr>
<tr>
<td>FRC</td>
<td>Fourier Ring Correlation</td>
</tr>
<tr>
<td>FSC</td>
<td>Fourier Shell Correlation</td>
</tr>
<tr>
<td>GIoU</td>
<td>Generalised Intersection over Union</td>
</tr>
<tr>
<td>GPU</td>
<td>Graphical processing unit</td>
</tr>
<tr>
<td>HAADF</td>
<td>High-angle annular dark field</td>
</tr>
<tr>
<td>IoU</td>
<td>Intersection over Union</td>
</tr>
<tr>
<td>IS</td>
<td>In-situ</td>
</tr>
<tr>
<td>k</td>
<td>Coulomb constant</td>
</tr>
<tr>
<td>$k_B$</td>
<td>Boltzmann constant</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>l(·,·)</td>
<td>SSIM luminance term</td>
</tr>
<tr>
<td>L</td>
<td>Length</td>
</tr>
<tr>
<td>LED</td>
<td>Local extension deblur</td>
</tr>
<tr>
<td>LP EM</td>
<td>Liquid-phase electron microscopy</td>
</tr>
<tr>
<td>LSTM</td>
<td>Long short-term memory</td>
</tr>
<tr>
<td>LTEM</td>
<td>Liquid transmission electron microscope</td>
</tr>
<tr>
<td>m_e-</td>
<td>Electron mass</td>
</tr>
<tr>
<td>MIT</td>
<td>Massachusetts Institute of Technology</td>
</tr>
<tr>
<td>ML</td>
<td>Machine learning</td>
</tr>
<tr>
<td>MOS</td>
<td>Metal Oxide Semiconductor</td>
</tr>
<tr>
<td>MOSFET</td>
<td>Metal Oxide Semiconductor Field Effect Transistor</td>
</tr>
<tr>
<td>MRI</td>
<td>Magnetic resonance imaging</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean-squared error</td>
</tr>
<tr>
<td>MSAD</td>
<td>Mean squared angular displacement</td>
</tr>
<tr>
<td>MSD</td>
<td>Mean squared displacement</td>
</tr>
<tr>
<td>n</td>
<td>Refractive index</td>
</tr>
<tr>
<td>N2N</td>
<td>Noise2Noise</td>
</tr>
<tr>
<td>N2V</td>
<td>Noise2Void</td>
</tr>
<tr>
<td>NN</td>
<td>Neural</td>
</tr>
<tr>
<td>NMR</td>
<td>Nuclear magnetic resonance</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal component analysis</td>
</tr>
<tr>
<td>PDI</td>
<td>Polydispersity index</td>
</tr>
<tr>
<td>PID</td>
<td>Progressive image denoising</td>
</tr>
<tr>
<td>PSF</td>
<td>Point spread function</td>
</tr>
<tr>
<td>PSNR</td>
<td>Peak signal-to-noise ratio</td>
</tr>
<tr>
<td>r</td>
<td>Distance between charges</td>
</tr>
<tr>
<td>R</td>
<td>Particle radius</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>------------</td>
</tr>
<tr>
<td>ReLU</td>
<td>Rectified linear unit</td>
</tr>
<tr>
<td>RGB</td>
<td>Red, green and blue</td>
</tr>
<tr>
<td>RMSD</td>
<td>Root mean squared displacement</td>
</tr>
<tr>
<td>ROI</td>
<td>Region of interest</td>
</tr>
<tr>
<td>s(·,·)</td>
<td>SSIM structural term</td>
</tr>
<tr>
<td>SEM</td>
<td>Scanning electron microscope</td>
</tr>
<tr>
<td>SIFT</td>
<td>Scale-invariant feature transform</td>
</tr>
<tr>
<td>SGD</td>
<td>Stochastic gradient descent</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal-to-noise ratio</td>
</tr>
<tr>
<td>SPA</td>
<td>Single particle analysis</td>
</tr>
<tr>
<td>SSD</td>
<td>Solid-state drive</td>
</tr>
<tr>
<td>SSIM</td>
<td>Structural similarity index measure</td>
</tr>
<tr>
<td>std</td>
<td>Standard deviation</td>
</tr>
<tr>
<td>STEM</td>
<td>Scanning transmission electron microscope</td>
</tr>
<tr>
<td>SVM</td>
<td>Support vector machine</td>
</tr>
<tr>
<td>T</td>
<td>Temperature</td>
</tr>
<tr>
<td>TEM</td>
<td>Transmission electron microscope</td>
</tr>
<tr>
<td>q_e^−</td>
<td>Electron charge</td>
</tr>
<tr>
<td>q_i</td>
<td>Electric charge (i-th element)</td>
</tr>
<tr>
<td>v</td>
<td>Velocity</td>
</tr>
<tr>
<td>V_a</td>
<td>Applied voltage</td>
</tr>
<tr>
<td>V_clk</td>
<td>Clock voltage</td>
</tr>
<tr>
<td>v_e^−</td>
<td>Electron velocity</td>
</tr>
<tr>
<td>V_h</td>
<td>Heating voltage</td>
</tr>
<tr>
<td>V_G</td>
<td>Gate voltage</td>
</tr>
<tr>
<td>V_{GS}</td>
<td>Gate-source voltage</td>
</tr>
<tr>
<td>x_i</td>
<td>Input signal (i-th element)</td>
</tr>
</tbody>
</table>
LIST OF ABBREVIATIONS

\( y \) Output signal

\( \alpha \) Aperture angle

\( \eta \) Fluid viscosity

\( \theta \) Rotational angle

\( \lambda \) Wavelength

\( \rho \) Resolution

\( \sigma \) Covariance

\( \varrho \) Fluid density

\( \omega_i \) Weight (i-th element)

CPEB4-D4 Cytoplasmic polyadenylation binding element

NH\(_3\) Ammonia

PBS Phosphate buffer solution

PEG-PDPA Poly(ethylene glycol)-poly(2-(diisopropyl amino) ethyl methacrylate)

PEG-PMET Poly (ethylene glycol)-b-poly(L-methionine)

PMPC-PDPA Poly(2-(methacryloyloxy)ethyl phosphorylcho-line)-poly(2-diisopropyl aminoethyl methacrylate)

PTA Phosphotungstic acid

SiN\(_2\) Silicon nitride

SiO\(_2\) Silicon oxide

\( \wedge \) Cross-product

\( \ast \) Convolution
CHAPTER 1

INTRODUCTION

Electron microscopy (EM) is an imaging technique used to obtain high resolution images of matter down to the atomic level [1], exploiting the interaction between the beam of electrons generated by the microscope and the specimen. EM is possible only by keeping the beam under vacuum in order to avoid undesired scattering events in the electron path, and thus allowing electrons to interact strongly with the specimen under study [2]. EM samples are conventionally imaged in the solid state. Imaging samples in liquid or containing liquid requires special preparation techniques involving either controlled drying or cryogenic treatments. Unfortunately, drying techniques may damage the specimen, or may corrupt the imaging process, introducing artifacts or facilitating the creation of aggregation of particles [3]. Such alterations become particularly critical for biological samples and soft materials, the structures of which comprise water as one of the main building blocks. Some of these limitations can be overcome by using fast vitrification processes to solidify liquid samples [4], however vitrified water is not liquid water and its structure and hydrogen bond network is very different [5]. Nevertheless, in the last decades cryogenic electron microscopy (cryo-EM) has revolutionised the field of structural biology [6]. In particular, the development of single particle analysis (SPA) [7] has enabled the three-dimensional (3D) reconstruction of the density map and structure of proteins, starting from their two-dimensional (2D) projections. SPA has had a terrific impact in the field of drug discovery, since structural information can be correlated with function [8].

In the last decades, the technological progress favoured the development of liquid-phase electron microscopy (LP EM), a technique that offers the possibility to image samples in their liquid native state [9]. The pivotal element of LP EM is the liquid-cell, an enclosure realised of electron transparent materials that entraps samples in liquid state, without interfering with the electron beam of the microscope [10]. The birth of LP EM enables the possibility to image samples in their liquid native state, and has been widely used to monitor in-situ electrochemical reactions [11]-[12], and the growth of nanocrystals [13]-[14], amongst many other fields. The present thesis exploits LP EM to observe objects
at nanoscale moving freely in their liquid native state. The images recorded via liquid transmission electron microscope (LTEM) are processed in order to derive information about the structure and the dynamics of objects dispersed in the liquid media. Towards this aim, new computer vision approaches have been designed accounting for the presence of liquid in the specimen, and the motion and dynamic events of the dispersed objects (i.e. particles), all elements that are extraneous to conventional EM.

Every component behind the instrumental settings for performing LP EM has been designed and adapted to the specific needs of liquid imaging, with the aim of trying to minimise the beam damage of the specimen and ultimately alterations of its structure. Two of the most important components of the instrumental set up are the liquid holder comprising a liquid cell, and a high resolution direct detection camera. The liquid holder employed in the present investigations was manufactured by DENSsolutions and presents a liquid cell where the sample is placed. The liquid cell is comprised of two microchips (top and bottom), with 50nm thick electron-transparent window each at the geometrical centre made of SiN$_2$, which does not interact with electrons. The two microchips form a microfluidic channel that keeps the sample hydrated. Furthermore, the cell is placed at the tip of the sample holder and sealed to prevent leaks. The top and bottom microchips are separated by spacers with variable thickness, from a minimum of 200nm up to a maximum value of 5µm, in order to image samples of different sizes.

The camera used to perform imaging in liquid is the Gatan K2-IS with capability for high definition image acquisition, up to about 4000 pixels for side; and temporal resolution, up to 1600 frames per seconds (fps). The camera works via a direct detection device (DDD), a technology that identifies the electrons as soon as they hit the detector. In this fashion, the noise generated by the electronic components is reduced, thus guaranteeing a very high detection quantum efficiency (DQE), the squared ratio of output signal-to-noise ratio (SNR) over input SNR [15]. The association of the LTEM with the K2-IS camera creates a system with the capability to image soft particles in their liquid native state without altering their structure and record their evolution in time. Soft materials such as micelles and vesicles form in water as a result of counteracting interactions, such as hydrophobic effects, hydrogen bonds and electrostatic interactions. Such relatively weak forces give rise to highly dynamical assemblies that exist in solution. Thus, LP EM is becoming the ideal tool to study the evolution and formation of vesicles and micelles in
solution. In our investigation, we exploited LP EM to image the dynamics of particles undergoing Brownian motion, using their natural rotation to access all the particle views in order to reconstruct their 3D structure by using tomographic techniques. In order to maximise the performances of the imaging process, the exposure time of the specimen to the beam has to be controlled, in order to prevent damage to the structure under investigation. Moreover, the frame rate of the camera has to be set up and tailored to each experiment and specimen in order to avoid motion blur, a side effect occurring while imaging moving objects [16].

The results of the imaging process are then elaborated via specific digital tools, designed around the limitations of LP EM. Conventionally, EM images results to be heavily affected by different types of noise, mainly due to secondary electrons, the fluctuation in the intensity of the beam, and the readout noise of the detector [17]. Moreover, LP EM adds an extra component to the overall existing noise because of the liquid nature of the specimen. Consequently, different deblurring and denoising approaches were adopted to improve the SNR of the images.

Furthermore, a big portion of this thesis is dedicated to the reconstruction of the 3D model of the samples imaged in liquid-phase. This task was performed by developing two different methods: Brownian tomography (BT) and Brownian particle analysis (BPA) [18]. The former method tracks in time a single particle that has been imaged via LP EM, recording all the rotational and translational movements. Thus, the particle shows to the camera many different profiles, providing the reconstruction algorithm with a rich set of orientations of the particles, needed to reconstruct the 3D model of the sample with great accuracy. Conversely, BPA is an extension in time of the SPA technique. Conventionally, SPA has been widely paired to cryo-EM to reconstruct 3D density maps starting from thousands of EM images by capturing hundreds of particles of the same species frozen on a grid. The SPA technique then extracts from these images the 2D profiles of particles to feed to the 3D reconstruction algorithm. On the contrary, BPA has the ability to process videos that may not contain thousands of particles, but instead monitors individual particle views across many consecutive frames, rather than across a single frame. In this fashion, the orientations needed for a successful 3D reconstruction are produced by recording the motion of the particles in solution over time. The path leading to the reconstruction of 3D structure of the herein studied specimen, mostly organic soft materials, starting from LP
EM videos is comprised of many steps. Briefly, the images are processed to increase the SNR and recover structural features hidden by the noise. The different profiles projected by the imaged samples are extracted, and ordered by angular displacement. Once the alignment is completed, all the features in the images have to be mapped in a 3D map, generating the 3D model.

In this fashion, a novel method to reconstruct 3D density maps of protein in liquid solution is defined. The great advantage of this approach consists in assigning the time factor a fundamental role in the reconstruction process. Consequently, merging LP EM and 3D reconstruction technique may pave the way to new frontiers in soft matter and structural biology. Thus, functional studies can be expanded to include dynamics, by creating a four-dimensional (4D) domain (i.e. 3D plus time) that enables understanding conformational changes of the samples, interactions between particles, or role of water in protein structure.

1.1 Thesis organisation

This thesis is structured in chapters as follows:

Chapter 2 An overview of the state-of-the-art of EM, the physics supporting it, and the algorithms involved in the post-processing analysis.

Chapter 3 A description of the methodology adopted to tackle the limitations of the LP EM technique, such as noise and blur, and the approaches implemented to enhance the quality of LP EM performed on organic materials. An explanation of the developed algorithms is provided, and different adopted strategies are analysed.

Chapter 4 The results obtained in this thesis are presented and discussed, together with the problems deriving from the adopted approaches.

Chapter 5 The conclusions of this research project are provided, together with suggestions for future improvements of the project.
2.1 **Electron Microscopy**

Microscopy is the act of using microscopes to image objects that the human eye is not able to see. There are three main branches of microscopy: optical, electron and scanning probe [19]. Even if the branches of microscopy differ in the technology used, the working principle is common to all of them: a source comprising radiation, electron beam, or a physical probe, respectively, is oriented and scanned over a sample, in order to create an image through a detector [20]. The technology of the microscope, the architecture of the system designed to perform the imaging process, and the nature of the sample to image vary according to the type of analysis, and the desired resolution [19].

Electron microscopy (EM) is a very specialised technique born in the first decades of the 20th century with the invention of the electron microscope [21]. This revolutionary tool magnifies patterns at the nanoscale – down to the atomic level – and transforms them into images [22]. Several type of electron microscopes have been developed, tailoring the design around different imaging techniques. However, all type of electron microscopes work in a similar fashion: an electron gun produces, focuses and accelerates a stream of electrons, which travels through a column, where it is further focused and deflected by a system of magnetic lenses. The beam of electrons hits the sample placed on the holder, and an image is created as result of the interaction between the electrons of the beam and the sample [23]. In the following sections, different types of electron microscopes will be presented in detail.

2.1.1 **The electron microscope**

The first element comprising an electron microscope is the electron gun. This device is placed at one end of the column and is responsible for generating the electron beam. Electron guns can be manufactured according to two main different architectures: heated filament (*Figure 2.1a*) or field emission gun (*Figure 2.1b*) [24]-[25]. The former is made of a cathode (generally a tungsten filament) heated by a low voltage (\(V_h\)) that generates a cloud of electrons due to thermionic effect [26]. The electron cloud is further focused by a
Figure 2.1: Schematic diagram of the two electron guns technologies. a) Heated filament: a cathode (in red) is heated at low voltage $V_h$ generating an electron cloud, further focused by an electrode and accelerated towards the anode by a high voltage $V_a$. b) Field emission gun: a rod-shaped cathode with a negative potential with respect to the anode ($V_a$) generates an electric field that emits electrons, if strong enough. The electrons are further focused into a beam by the anode.

Negatively charged electrode (e.g., a Welnelt cylinder), and accelerated towards the anode by a high voltage ($V_a$) applied between the hot cathode and the anode [24]. Conversely, a field emission gun (FEG) has a rod-shaped cathode with a very thin tip, held at a negative voltage ($V_a$) with respect to the anode potential, thus generating an electric field. If this field is strong enough, it is possible to have the electrons spontaneously emitted. Then, the anode focuses the electrons into a beam. The FEG technique is able to produce a beam of electrons much more powerful and narrower in diameter than the beam obtained with the heated filament [25].

The lenses are magnetic multipoles made of copper coil windings, that generate a magnetic field when run by an electric current. The intensity of the electric current and the number of windings determine the intensity of the magnetic field, which then interacts with the electrons in the beam running through [27]. A schematic representation is depicted in Figure 2.2. The lens-electrons interaction can be modelled by the magnetic Lorentz force ($F_L$) (Equation 2.1).

$$F_L = -q_e\cdot(v_e \times B)$$

where $q_e$ is the charge of the electrons, and $\times$ the cross-product between their velocity vector ($v$) and the magnetic field vector ($B$) [28]. After passing through the magnetic lenses, the electron beam collides with the specimen, where the electrons interact with
the atoms of the sample. The interaction between the beam and the sample makes the electrons of the beam deviate from their own original trajectory [29]. This phenomenon (also known as scattering) is modelled by the Coulomb’s law, which measures the force between two electrical charged particles (Equation 2.2) [29].

\[ F_C = k \frac{q_1 q_2}{r^2} \]  

(2.2)

The intensity of the force \( F_C \) modelling the interaction between two charges is directly proportional to Coulomb’s constant \( (k) \), the magnitude of the charges \( (q_1 \) and \( q_2) \), and is inversely proportional to the squared value of the distance between them \( (r) \). The way the atoms of the sample scatter the electrons of the beam is depicted in Figure 2.3, a representation that follows the Bohr model of the atom [30]. This model includes a positively charged nucleus surrounded by negatively charged particles (i.e. the electrons) travelling in circular orbits, with quantified energy levels, proportionally to the orbit size. The smallest orbit has the lowest energy. Moreover, electromagnetic energy can be absorbed or emitted if electrons move into different orbits [30].

The incoming electrons can experience multiple scattering events as they can interact with the electrons shell or the nucleus of the atoms of the specimen. The first type of scattering event, generally implies inelastic collision. In inelastic scattering part of the kinetic energy of the incoming electrons of the beam is transferred to the specimen. In some cases, this transferred energy may excite an outer electron of the atomic shell and the excited electron is ejected into the vacuum. These electrons are also known as secondary electrons. The energy loss may generate emission of energy in form of X-rays or photons. Furthermore, in inelastic scattering the momentum is not conserved and a slight deviation in the path
Figure 2.3: Schematic diagram showing the process of incoming electrons interacting with a single atom of a specimen, depicted using Bohr’s atomic model. Inelastic scattering occurs if the collision between one electron of the beam and an electron of the atom of the specimen does not conserve kinetic energy, thus slightly modifying the trajectory of the incoming electron. Moreover, the energy loss can generate emission of energy in form of X-rays or photons, or can be transferred to the electron of the atom, modifying the chemical bonds and thus producing secondary electrons. Conversely, elastic scattering event between the electron of the beam and the atom of the specimen have conservation of kinetic energy. Elastic scattering may occur because of the attractive force of the nucleus, which deflects the path of the incoming electron.

of the incoming electron of the beam may take place. On the contrary, the scattering events taking place between the incoming electrons and the nucleus is mostly elastic, with no energy losses. In this case, the incoming electron is attracted by the positively charged nucleus and deviated from its original path. In some cases, the deviation is so intense that electrons may be backscattered [29]. After interacting with the atoms of the specimen, the electrons of the beam are collected by a detector, which converts the energy of the electrons into an image [22]-[31].

The detector is another pivotal device in the setting of an electron microscope, as it is responsible for transforming the electrons into digital signals, which are further transformed into images. Initially, images were recorded on photographic film, which for electron microscopy consisted of a silver halide emulsion layer on a plastic support. After the exposure to the incoming electrons, the film was chemically developed forming an image [32]. However, starting from the 1980s, the photographic films were replaced with digital detectors, able to transform the signal carried by the electrons into a digital image. Regarding the technology involved in the design of the detectors for EM, it is worth mentioning that the architecture of these devices can be grouped into two main categories:
charge-coupled device (CCD) and direct detection device (DDD). The former is an integrated circuit that consists of a matrix (i.e. a 2D array) of capacitors (i.e. the pixels) that accumulate and transfer the incoming electrons. These electrons are further transformed into a voltage value and lastly in a digital number by an amplifier connected down-stream to the matrix. In order to understand how CCD fully works, it is important to analyse the way the pixels are manufactured [33]. These elements are capacitors formed as a “sandwich” structure of metal oxide semiconductor (MOS). Generally, the semiconductor substrate is p-doped (i.e. doped with positive charges, or “holes”), while the metal (also called “gate”) is biased at a positive potential ($V_G$).

![Schematic representation of the working principle of a CCD](image)

*Figure 2.4: Schematic representation of the working principle of a CCD. a) A pixel of the detector is defined as a metal oxide semiconductor sandwich structure, named MOS. The surface (also known as “gate”) is positively polarised by the gate voltage ($V_G$), which forces the negative charges occupy a depletion region (in yellow) close to the oxide, while the positive charges move down to the bottom. The photons hit the gate and produces electrons by photoelectric effect that are stored in the depletion area. b) A clock voltage ($V_{clk}$) is applied to a matrix of CCD transforming the detector into a shift register. The difference of potential creates a conduction channel (in yellow), having the charges flow from a MOS to the next one.*

In this fashion, all the holes move away from the gate, where a depletion area is created and filled with negative charges (*Figure 2.4a*). Thus, this electrical state of the MOS is named *depletion mode*. At this point, when the electrons of the beam hit the surface of the MOS, electrons are generated by photoelectric effect [34] and stored in the depletion well below the surface. After collecting the charges during the exposure time, a clock signal ($V_{clk}$) (i.e. voltage) is applied to the rows of the CCD, driving the device as a shift register, i.e. a device shifting the signal from a component to the next one at each event of the clock
A positive voltage is applied to the gate of two consecutive rows at a time, in order to create a conduction channel between two adjacent MOS devices, allowing the charges to flow from one capacitor to the next one (Figure 2.4b). Once the charges are transmitted down to the last row of the matrix, they are set in input to a charge amplifier that transforms the charges into a voltage. This analog signal is further converted into a digital value by the analog-to-digital converter (ADC) connected downstream.

CCDs are robust detectors widely used in EM. Unfortunately, this architecture presents some strong limitations, as the ability to absorb only low energy photons and the reduced field-of-view, factors that lower the efficiency of the device. A possible solution to overcome this problem relies in adopting a scintillator, a coating for wavelength conversion that allows the detector to absorb photons of high energy. This technique is widely adopted as it increases the signal-to-noise (SNR) [36]. However, adopting a DDD as detector may represent a solution to overcome some of the limitations of the CCD. The DDD class of detectors is based on complementary metal oxide semiconductors (CMOS) that amplify the signal into each pixel, directly converting charges to voltage very efficiently, and thus enabling fast acquisition [37]. This circuit consists of two MOS Field Effect Transistors (MOSFET) connected together (Figure 2.5). The MOSFET adds to the MOS architecture two new terminals, named “source” and “drain”. Moreover, a voltage is applied between gate and source to control the current flowing from the source to the drain. These two new terminals are doped with charges with opposite polarity with respect to the substrate, and the polarity of the dope of these terminals names the transistor: nMOSFET and pMOSFET, if the terminals are negatively or positively doped, respectively. A CMOS sensor is realised by creating a circuit that connects both gate terminals together and the drain of an nMOSFET to the source of a pMOSFET. Different types of dope for gate and drain have the MOSFET respond to opposite voltage in input. This circuit design acts as a switch and thus allows one of the MOSFET to be always...
in conduction, letting the CMOS detector working at a very wide range of input voltage. In addition, CMOS detectors have amplifiers connected to every pixel that transform the photons into voltage values, thus significantly reducing the processing time. Furthermore, since the electrons are identified as soon as they hit the detector rather than being first integrated (as in CCD cameras), the Landau noise and the readout noise are significantly reduced. The former is due to energy deposited on the detector at every electron event [38]; while the latter is produced by the electrical components of the camera detector [39]. Decreasing the effect of both sources of noise increases the detection quantum efficiency (DQE), which expresses the squared ratio of output SNR over input SNR [15]. Consequently, DDD-based detectors have been widely adopted in EM, progressively replacing CCD-based detectors. The DE-64 manufactured by DirectElectron has been used in cryo-EM [40]; the Falcon line of detectors (I, II and 3EC) from Thermo Fisher has been used to collect high quality data for crystallography [41], cryogenic electron microscopy (cryo-EM) of biological macromolecules [42], and 3D reconstruction via SPA [43]. In addition, Gatan manufactured the K2, which is adopted to record the images present in this thesis, and the K3 [44]. Both cameras have been used to perform super-resolution imaging of crystals [44], low-dose imaging [45], and widely for reconstruction of samples imaged via cryo-EM [46]-[47]-[48].

However, despite the architecture of the selected detector, the EM imaging process is strongly limited mainly by three factors (Figure 2.6):

- **high vacuum condition**: the presence of particles in the path of the beam of electrons may cause some undesired scattering events [2]. Consequently, the final EM image would show some artifacts (i.e. features appearing in the image, but missing in the original sample), negatively affecting further analysis (Figure 2.6a).

- **beam damage**: during long-time exposure the imaged sample may absorb an incredible amount of energy from the beam. This phenomenon significantly alters the chemical bonds in the atoms, deeply modifying the chemical and structural properties of the specimen. Beam damage can cause scission, mass loss and ionisation, amongst others (Figure 2.6b) [49].

- **spherical aberration**: a perfect lens makes all the rays passing through converge to the same focus point. However, a spherical aberrated lens cause rays at different angles focus in different points, thus producing distinct foci and consequently a blurred
Nevertheless, the spherical aberration can be compensated by installing in the microscope a corrector system [51].

These three factors strongly limit the quality of the imaging process, consequently, heavily affecting the resolution of the imaged samples. Resolution can be defined as the smallest measurable distance between two points and is modelled by the Rayleigh criterion for optical systems (Equation 2.3) [52].

$$\rho = 0.61 \frac{\lambda}{n \sin(2\alpha)}$$  \hspace{1cm} (2.3)

where \(n\) is the refractive index of the medium between the lens and the sample, \(\alpha\) is half of the angular aperture (i.e. the angular size of the lens from the focal point) and \(\lambda\) the wavelength of the observed radiation. In case of EM, \(\lambda\) parameter is proportional to the voltage applied to generate the beam of electrons via the de Broglie equation, which relates the wavelength (\(\lambda\)) to the Plank constant (\(h\)), the mass of the electron (\(m_e\)), its charge (\(q_e\)) and the applied voltage (\(V_a\)) (Equation 2.4) [52].

$$\lambda = \frac{h}{\sqrt{2m_e q_e V_a}}$$  \hspace{1cm} (2.4)
As previously stated, there are different types of electron microscopes as well as imaging techniques. A brief introduction to the three main types of electron microscopes and imaging techniques are discussed [53]. A graphical comparison is provided in Figure 2.7.

2.1.2 Transmission electron microscope

Transmission electron microscopy technique owes its name to the strategy involved to perform the imaging process. A beam of electrons is generated and transmitted through a thin sample (i.e. below 100nm), and then collected by a detector, which transforms the electrical signal into an image in output [54]. Once the beam is generated, it is then focused and oriented by the system of lenses (as previously explained) towards a thin sample (Figure 2.7a). The limited thickness is a strong requirement a sample has to satisfy, in order to produce high quality images. As discussed in Subsection 2.1.1, the electrons of the beam strongly interact with the atoms of the specimen, resulting in multiple scattering events, during which energy and momentum may be transferred to the specimen. Consequently, the outcoming electrons reaching the detector may contain very low level of energy. Finally, the detector is responsible for measuring the energy contained in the outcoming electrons and converting it into a digital image [55].

2.1.3 Scanning electron microscope

A scanning electron microscope (SEM) is characterised by a beam of electrons focused in a fine spot, which is further scanned in a raster way over a sample. The beam is oriented by the coil deflecting system, an electromagnetic structure similar to the lenses, that deviates the beam over the sample. The electrons of the beam experience multiple scattering events, thus generating back scattered electrons in the case of elastic scattering events, secondary electrons and often x-rays, as consequence of inelastic scattering events. The outcome of these scattering events are collected by a specific detector, concurring to the formation of digital images. Conventionally, SEM images are generated by elaborating the information carried by the low energy secondary electrons (Figure 2.7b) [56]. Consequently, the produced images generally reach worse resolution than TEM, but enable imaging bulk materials.
2.1.4 **Scanning transmission electron microscope**

A scanning transmission electron microscope (STEM) merges the working principle of both TEM and SEM, by adding a deflecting system to the TEM architecture. In STEM, a thin sample is scanned in a raster fashion and the electrons coming through it are collected by a detector (Figure 2.7c). The detectors arrangement of the STEM consist of an annular detector recording electrons elastically scattered around the path of the electron beam. This imaging mode is consequently known as annular dark field (ADF) [57]. Furthermore, it is possible to create high quality images by collecting electrons highly deviated from the original trajectory, by implementing a high-angle ADF (HAADF) detector. The adoption of this type of detectors enables the creation of images in which the contrast is proportional to the atomic number of the element in the specimen [58]. Moreover, STEM allows also the collection of axially transmitted electrons, by the bright field detector located in the path of the electron beam. In case of bright field images, the contrast is proportional to the mass and the thickness of the sample, as in case of conventional TEM. Therefore, often bright field and HAADF images are intended as complementary, thus providing complete information of the imaged specimen [59].

![Figure 2.7: Schematic representation of the main EM techniques.](image)

**Figure 2.7:** Schematic representation of the main EM techniques. **a)** TEM focuses the electron beam over a thin sample and the image is created by collecting the electrons coming through it. **b)** SEM focuses the electron beam in a very small spot and scans it over a thick sample in a raster way. The image is created by collecting secondary electrons and back-scattered electrons. **c)** STEM merges the other two techniques, by scanning a thin sample and creating the image with the electrons coming through.

2.2 **Sample Preparation**

In the previous sections the importance of the vacuum-condition and the dependency of the three most common EM techniques to the thickness of the sample were highlighted.
In order to satisfy the requirements necessary to image a sample, several preparation techniques have been developed over the years. The adopted technique have to be tailored to the nature of the specimen and the parameters to investigate in the imaged specimen. Two of the most common methods are positive and negative staining, and cryogenic electron microscopy (cryo-EM), amongst others [60].

2.2.1 Staining

Negative staining was firstly introduced in the late 1950s to study the structures of virus, which were challenging to image due to the low contrast generated during the imaging process [61]. The staining technique overcomes this limitation and facilitates the EM analysis of organic samples, which generally produce low contrast during the imaging process, due to low atomic number and density similar to surrounding media [62]. The negative staining technique consists in embedding the particles of the sample on a grid in a thin layer of electron dense material (e.g. phosphotungstic acid (PTA), uranyl acetate, or uranyl formate), which dehydrates the area surrounding the particles and sustains them. The difference in electron density between the sample and the staining material translates in high contrast during the imaging phase, having bright particles surrounded by dark supports [63]. Conversely, the positive staining is absorbed by the sample but not by the background, thus colouring the particles and enhancing the contrast with respect to a bright background. Therefore, the positive staining technique is adopted to identify and measure the organic samples [64], while negative staining is useful to recover structural information of the imaged sample [61]. Negative staining has the great advantage to be easy to prepare, and last long, thus the grids with the sample can be re-imaged. Unfortunately, it may create artifacts, damaging the sample and altering the produced images. However, the greatest limit introduced by negative staining concerns the resolution. The negative stain adopted to coat the sample is comprised of nanoparticles having average size of 5Å to 7Å. These grainy particles cover the sample, hence representing the smallest object that can be resolved [65]. Consequently, the smallest resolution value that can be obtained results to be about 20Å, according to the Nyquist-Shannon sampling theorem [66].
2.2.2 Cryo-EM

Cryo-EM is a very popular sample preparation technique that preserves via vitrification the specimen in a state close to native and enhances the resistance of the sample to the beam damage [67]. This technique requires the sample to freeze incredibly fast, fixing the surrounding environment in vitreous state and avoiding the formation of crystalline ice structures. The presence of vitreous ice structures compromises the integrity of the samples, thus degrading the quality of the images [68]. Cryo-EM can be implemented by following two different procedures [60]:

- **plunge freezing**: a very thin layer of specimen in liquid state is plunged into a cryogenic agent (such as liquid ethane or propane) and quickly frozen (less than 10ms). The sample has to be frozen at very high speed in order to avoid the undesired formation of ice structures, due to different values of temperature of the specimen. The plunge freezing technique is widely used, but unfortunately the specimen has to be imaged in low dose to avoid beam damage and sample melt [69].

- **high pressure freezing**: thick samples (i.e. between 100µm and 300µm) are frozen by increasing the pressure applied to the sample to about 2000bar and cooling down the temperature using liquid nitrogen. Unlike plunge freezing, high pressure freezing is adopted to prepare thick samples, which have to be sliced to be imaged via TEM [70].

Embedding the specimen in a cryogenic agent has the great advantage of preserving the particles of specimen from direct exposition to the electrons of the beam [71], thus reducing the effect of the beam damage by three or four order of magnitude [72]. Consequently, the advent of cryo-EM had a significant improvement on the resolution of EM images, as Dubochet demonstrated in 1988, when he recorded a series of high quality images embedded in ice [73]. *Figure 2.8* reports examples of the specimens imaged by Dubochet, who recorded viruses embedded in ice, and macromolecular complexes as ribosome and ferritin, all specimens producing good results when imaged with other classical methods [73]. A comparison between the result of the imaging process of a solution of ferritin protein fixed with both negative staining and cryo-EM is provided in *Figure 2.9.*

The capability of preserving samples in a state close to native and the increased resistance to the beam damage were two of the main causes of the impressive impact the
Figure 2.8: High quality cryo-EM images recorded by Dubochet in 1988, depicting Semliki forest virus (a), T₄ bacteriophages (b), ferritin and apoferritin (c)). Images taken from the original paper [73].

Figure 2.9: Comparison between the result of the imaging process of a solution of ferritin protein fixed with negative staining (a) and cryo-EM (b). Images are taken from the papers [73] and [74], respectively.

development of cryo-EM had in biochemistry. Especially, it revolutionised the reconstruction of 3D models of biological samples, taking the leading role from x-ray crystallography. In 2017, Dubochet, Frank and Henderson won the Nobel Prize for the development of cryo-EM technique [75].
2.3 3D Reconstruction

Acquiring detailed information about the structure of macromolecules is of great importance, since structural information can be associated with function [8]. The advent of cryo-EM technique represented a turning point in structural biology, as it enables the reconstruction of near-atomic structures, introducing different advantages with respect to x-ray crystallography and nuclear magnetic resonance (NMR), the only tools at that time able to reach atomic resolution [8]:

- x-ray crystallography and NMR require a great quantity of sample to analyse, in the order of several milligrams, while cryo-EM about 0.1mg.
- x-ray crystallography - as the name may suggest - requires the protein solution to crystallise before imaging it with a powerful x-ray beam. However, crystallising a protein is a very difficult task. Conversely, the specimen to image with NMR technique demands low concentration in order to produce a wide and rich spectrum of signals about the elements composing the specimen. Differently, the procedure of sample preparation for cryo-EM was discussed in the previous subsection.
- x-ray crystallography produces density distributions of the specimen after analysing the diffraction pattern with several methods for structural analysis and data fitting [76]. On the other hand, NMR leads to 3D structures of the specimen after applying different data processing tools to the recorded spectrum [77]. Conversely, cryo-EM requires computer vision algorithms to produce 3D density maps of the imaged specimen.

The diffusion of cryo-EM marked the dawn of a new era for biologists, which was further named resolution revolution [78]. Another great advantage introduced by the adoption of electron microscopy to perform 3D reconstruction relies on the much lower damage the electron beam produces when compared to x-rays. Henderson measured that x-rays produce 1000 times and 20 times more damage than electrons for elastic and inelastic events, respectively [79].

In the following sections, a brief overview of the most common 3D reconstruction techniques is provided.
2.3.1 Electron tomography

Electron tomography has been largely applied to conventional TEM to reconstruct the density maps of specimens imaged in solid-state [80]. This reconstruction technique is performed by tilting the sample holder by finite increments (Figure 2.10a) - usually regular - and acquiring images at each orientation (Figure 2.10b). All the 2D projections of the imaged specimen are then aligned and the 3D model is reconstructed by applying computer vision software (Figure 2.10c). The reconstruction process is based on the central slice theorem (also known as Fourier slice theorem), which was initially proposed for 2D functions, but it can be easily generalised to N dimensions [81].

Generally, it is possible to map a generic function in the time domain into the Fourier space by simply applying the Fourier transform. However, the central slice theorem enables the recovery of the Fourier transform of a function, by starting from a collection of Fourier transforms in a low dimensional space. Thus, the Fourier slice theorem is very useful in the case of tomography reconstruction, since the an a priori 3D structure of the imaged specimen is unavailable, but it is possible to obtain a high number of 2D projections at different angles.

Here, the Fourier slice theorem in two dimensions is described for simplicity (Figure 2.11). The central slice theorem states that the one-dimensional (1D) Fourier transform of a projection of a 2D function obtained from certain angle ($\theta$) corresponds to a line of the Fourier transform of the entire object. In detail, the line (in red in Figure 2.11) passes through the origin of the Fourier domain at the

![Figure 2.10: Schematic diagram showing the working principle of conventional electron tomography. a) The sample holder hosting the specimen in solid state is tilted for a limited set of discrete orientation under an electron beam. In light blue the “missing wedge”, due to the structural limitation of the TEM. b) Different snapshots of the sample are taken at different orientations. c) The 2D pictures are used to reconstruct the 3D model of the sample.](image)
exact angle $\theta$. Therefore, the entire Fourier transform can be constructed by acquiring projections at multiple angles.

Finally, the original 2D function can be recovered by applying the inverse Fourier transform. Similarly, the 3D model of the imaged sample is recovered by applying the Fourier slice theorem on the set of 2D projections. Consequently, it is easy to understand that the quality of the reconstructed model is deeply connected to the number of 2D projections and the value of the sampling angle. Unfortunately, the set of available projection angles is limited to $\sim \pm 70^\circ$ due to the increasing thickness of the sample at high angles [82] and structural limitations. For example, the specimen chamber may be too small to accommodate a full rotation of the holder [83]. Usually, sample profile images corresponding to a holder rotation from $70^\circ$ to $90^\circ$ and $-70^\circ$ to $-90^\circ$ can not be taken, consequently leading to a lack of information, known as missing wedge (Figure 2.10a). The problem of the missing wedge can be overcome by tilting the sample on different axes - which requires a specific hardware setting -, or reconstructed via software interpolation [84]. Moreover, the central slice theorem samples the observed object with more points at low frequency (i.e. the center of the Fourier domain) than at high frequency. This differential sampling frequency translates into a lack of details - i.e. high frequency information -, blurring the final reconstructed model. However, this problem can be overcome by recurring - once more - to interpolation. The interpolated points are mapped from a polar to a Cartesian grid, before applying the inverse Fourier transform (Figure 2.12) [85].
Figure 2.12: Schematic representation of the interpolation operation developed to overcome the differential frequency sampling scheme of the central slice theorem. **a) The central slice theorem slices the Fourier space with lines passing through its center at different angles. Consequently, the sampled points are denser at low frequency (the center) than high frequency. b) Interpolating the polar space creates similar level of accuracy between low and high frequency in Cartesian space. The image is taken from the original paper [85].**

In the last decades, the electron tomography technique has been widely used [86] to study a vast number of polymeric structures, such as block copolymers [87] and nanocomposites [88]; and to reconstruct 3D models of organic specimens, with nanometer-resolution [89]-[90]. In Figure 2.13 an example of 3D restructured model obtained via electron tomography is provided. Moreover, the development of new computer vision algorithms together with new techniques have significantly improved the performance of the electron tomography technique. For example, the introduction of aberration corrected microscopes have pushed the resolution of the reconstructed 3D models to sub-nanometers level [91].

### 2.3.2 Single particle analysis

Electron tomography has shown to achieve great results in reconstructing 3D models, but it requires very specific hardware settings. One technique that actually dominates the scene of 3D reconstruction and that has been adopted more and more in the last decades is single particle analysis (SPA) [92]. SPA is an ensemble of computer vision algorithms applied to a series of images recorded via EM, showing multiple particles of the same species. The aim of these algorithm consists in reconstructing a high-quality 3D model of the imaged specimen. SPA can also be applied to stain-prepared samples, however it owes its great success to the development of cryo-EM sample preparation technique
and imaging. As already discussed in Subsection 2.2.2, the cryo-EM technique relies on vitrifying a large number of similar particles in their native state, which are then immobilised in a random orientation, showing different profiles to the camera during the imaging process. Thousands of images obtained with this procedure are fed to the SPA technique, which extracts all the profiles of the imaged particles and assign to each of them an angular orientation. Once all the profiles have been aligned, a 3D model (also referred to as “density map”) of the imaged particle is reconstructed (Figure 2.14) [93].
The first steps of the SPA processes the micrographs (i.e. the images recorded by EM) imaged via cryo-EM technique, compensating and reducing distortions and corruptions introduced during the imaging process. In order to reduce the beam damage effect, the samples are imaged at very low dose, with the consequent risk to produce images with very low contrast and SNR. One solution widely adopted to overcome the low contrast limitation consists in averaging a series of consecutive images of the same sample. In this fashion, the SNR of every micrograph used to feed the SPA is highly increased, thus producing high quality 3D models with near-atomic resolution [94]-[95]-[96]. Unfortunately, the micrographs to average may slightly differ due to the movements of the particles embedded in ice. Even if the particles are frozen on the grid, this phenomenon is sometimes inevitable. A mechanical drift of the microscope stage may induce a mono-directional drift. Another possible source of motion derives from the interaction between the beam and the sample, with a consequent transfer of energy to the particles [97]. All these factors cause the detector to register slightly different images. Averaging - even slightly - different images leads to a loss of details and smooth 2D projections, which further translate into blurred 3D models. Therefore, these variations of the particles position have to be compensated before averaging similar images. In order to fulfil this task, several software have been developed, Motioncorr [38] and Unblur [98] are considered to offer the highest performance.

After pre-processing the micrographs, the particles are ready to be selected from each image. It is easy to understand that manually selecting hundreds of thousands of particles may be time-consuming and tedious. Therefore, many semi-automated algorithms have been developed, such as RELION [99] and EMAN2 [100]. These methods require the software to create a template of the particle to search across all the images. Conventionally, the patches (i.e. portion of pixels of an image) identifying the particles are selected 1.5 or 2 times bigger than the actual size of the particle. In this fashion, the background is included in the process, facilitating the noise estimation and the correction of the contrast transfer function (CTF) [97]. The CTF is the Fourier transform of the point spread function (PSF), which represents how a single perfect point is imaged by the microscope. Thus, the PSF models the distortion introduced by the microscope, accounting for the spherical aberration and the defocus affecting the imaging process. Compensating these limitations enhances the accuracy of the final reconstruction [101].
SPA proceeds with the most computationally demanding step of the whole procedure, the assignment of the angular orientation to the profiles in the pre-processed images. The orientation assignment is worsened by the presence of noise and the low dose necessary to limit the beam damage [102]. A first solution to tackle the problem of assigning angular orientations consists in splitting the profiles in classes on a similarity criterion. Particle profiles showing similar details are then averaged to reduce the noise. Unfortunately, this approach significantly lowers the details, smoothing the peculiarities of every particle profile. Hence, this method is adopted to assign a first small set of angular orientations to the classes and thus generating a rough initial 3D model [103]. The majority of the algorithms developed to generate the initial 3D model are based on the common lines principle proposed by Crowther [104]. The common lines principle compares the sets of 1D projections derived from a pair of images, finding the projections sharing the same orientation (Figure 2.11). This approach is then applied to two classes at a time, finding the common lines and assigning the orientations to all the averaged classes.

The initial 3D model represents a good starting point for the assignment of the angular orientations to each profile. This task is the key to the success of the reconstruction, as incorrect assignments eventually result in a blurred model, consequently reducing the final resolution [105]. The initial model is then refined by applying iterative methods [106]-[107]. One of the most applied method is the projection matching algorithm that assigns three angular parameters - generally expressed as Euler angles [108] - defining a specific angular orientation, and two translational parameters to the center of every single particle profile that best matches the model projections [109]-[110]. From a mathematical point of view, the projection matching reduces to an optimization problem, minimising the distance \(d\) between the each profile \((I_{i=1,N})\) and the projections of the model \((R|\varphi_i,\theta_i,\psi_i)\) at the orientation given by the set of Euler angles \((\varphi_i, \theta_i, \psi_i)\) (Equation 2.5) [111].

\[
\arg\min_{\varphi_i, \theta_i, \psi_i} \sum_{i=1}^{N} d(I_i | R_{\varphi_i, \theta_i, \psi_i}) \tag{2.5}
\]

Many different tools have been developed to fulfil these last tasks of the reconstruction process: RELION [112]-[113]-[114], EMAN [115]-[116], IMIRS [117], FREALIGN [118]-[119]-[120] are the ones guaranteeing the highest performances, amongst many.

In the current section the main steps for the reconstruction of 3D models of samples imaged via EM are presented together with the corresponding image processing software.
implementation. However, it is important to highlight that all these packages are framed together into a single freeware named *Scipion* [121]. This framework allows the user to create a single workflow, which starts from inputting the EM micrographs and ends with the refined 3D model. A schematic representation of the workflow comprising all the steps is represented in *Figure 2.15*.

The reconstruction technique discussed in this section aims at recovering the 3D structure of the imaged specimen, starting only from the analysis of the acquired micrographs. This fashion is also known as *ab initio* reconstruction [122]. However, the reconstruction processes can be facilitated by adding as second input an existing 3D atomic model of the same sample obtained via x-ray tomography or crystallography [123]. The addition of this *a priori* model facilitates and improves the quality of the reconstructed model, as it constraints the angles assignment problem [124].

Having the correct model reduces the number of possible orientations that a 2D profile can occupy in space. A small gallery of successful SPA reconstructions is depicted in *Figure 2.16*. The micrographs and the consequent reconstructions have been obtained from the protein database website, selected amongst the proteins of the month section [89]-[125]-[126].
Figure 2.16: Small gallery of SPA reconstructions performed on specimens imaged via cryo-EM. In the top row the cryo-EM micrographs, while in the bottom row, the corresponding 3D reconstructions. a),b) 26s proteasome [89]. c),d) Human papillomavirus [125]. e),f) RNA-dependent RNA polymerase of SARS-CoV-2 coronavirus [126].

However, 3D reconstruction workflow may be affected by some structural flaws that reduce the goodness of the outcome [105]. Firstly, the well-known “model-biased” problem may occur when an incorrect initial 3D map is set in input to the refinement step and persists across the iterations. Another problem that may derive from misinterpretations of the data happens when the images fed to the algorithm do not contain signal (i.e. information), but only noise. Including pure noise in the process inevitably lowers the quality of the reconstruction [128]. The possibility to align noise to signal has been widely analysed and is known as “Einstein from noise”: if a dataset of random noise is aligned to a single image of Einstein, the noise features will align creat-

Figure 2.17: The results of the alignment of a random noise dataset with a single image of Einstein. The noise features align with the only image carrying information and create a noisy image of Einstein. Image taken from the original paper [127].
ing the image of Einstein [127]-[129] (Figure 2.17). Lastly, the “reinforced noise” consists in over-fitting the model, i.e. treating noise as high resolution features [130]. This last problem may lead to erroneous measurement of the resolution of the reconstructed model.

2.3.3 Resolution

The resolution is the key concept around which the 3D reconstruction techniques are tailored. The lower the resolution values, the more resolved and the higher the quality of the 3D reconstruction. The resolution value is deeply connected to the algorithm used to reconstruct the model and - of course - depends on the images fed in input. It has been proven that for images with high SNR, the resolution of a model can be estimated with the following equation [131]:

\[ \rho = D \theta \]  

(2.6)

According to Equation 2.6, the resolution (\( \rho \)) of a 3D model is proportional to the diameter (\( D \)) of the object to reconstruct and the value of the rotational increment (\( \theta \)). However, the progress in the computer vision field and the development of new technologies made this equation unreliable and produce only qualitative results. Nevertheless, the idea that the more refined the observation angle, the more the images and the lower the final resolution still persists. Therefore, an accurate measurement of the resolution of a reconstructed 3D model may express the goodness of the reconstruction itself [132]. In Figure 2.18 an example of the evolution of the resolution in the last few decades is provided, exploiting the reconstruction of ribosome from *escherichia coli* modelled in 1995 [133], 1998 [134], 2004 [135] and 2015 [136], respectively. During almost twenty years, the resolution of the reconstruction has been improved by a factor 8.

The algorithms developed to measure the resolution of 3D models can be grouped in two major groups [137]:

- techniques based on the comparison of averaged subsets of the data, as the Fourier Shell Correlation (FSC) [138], or the Differential Phase Residual (DPR) [139].
- algorithms based on the Fourier transform of individual images, as the Q-factor [140] and the spectral signal-to-noise ration (SSNR) [141].

The great advantage of the fist group of algorithms over the second one relies on the possibility to measure the resolution both in 2D and 3D [137].
Figure 2.18: An infographic about the evolution of the resolution of 3D reconstructions performed via SPA of ribosome from escherichia coli, imaged via cryo-EM. From left to right, 3D model reconstructed in 1995, 1998, 2004 and 2015, characterised with a resolution of 23 Å, 15 Å, 13.2 Å and 3 Å, respectively. Images taken from the original papers [133]-[134]-[135]-[136]

The method dominating the state of the art of the techniques to measure the resolution is by far the FSC. During the last years FSC has been adopted in so many different studies [142]-[143]-[144] to become the standard to measure the resolution for reconstructed 3D models. FSC was proposed in 1987 by Harauz and Van Heel and is derived as the 3D extension of the Fourier Ring Correlation (FRC). FSC measures the cross-correlation (i.e. a similarity measurement of two functions [145]) between two 3D models in the Fourier space (Equation 2.7) [146].

\[
FSC_{12}(r_i) = \frac{\sum_{r \in r_i} F_1(r) \cdot \overline{F_2(r)}}{\sqrt{\sum_{r \in r_i} F_1^*(r) \cdot \overline{F_2^*(r)}}}
\]  

(2.7)

where \( r \) is the spatial frequency, \( r_i \) is the value of the \( i \)-th voxel (i.e. volumetric pixel) in the Fourier domain, and \( F_1 \) and \( F_2 \) are two independent 3D models reconstructed by using different halves of the dataset. The bar over the \( F_2 \) represents the mathematical operation of the complex conjugate. The square root at the denominator has the function of normalising the result of the cross-correlation.

This equation compares equivalent regions of the two models with respect to frequency and determines as resolution the frequency at which the FSC drops below a specific threshold. Conventionally, this threshold is kept at 0.143, a value that is derived from the correlation between a reconstructed density map and a perfect reference map, obtained via x-ray tomography [147]. However the efficacy of FSC has been widely debated, and many other
methods were proposed to select a highly accurate threshold value [148]-[149]-[150]. Unfortunately, none of these methods may overcome some structural limitations introduced by the FSC itself. Firstly, the ratio behind splitting the dataset to create two different models inevitably biases the final resolution. Moreover, the FSC produces only a global value that does not take into account all the peculiarities of the reconstructed model. In order to overcome these problems, the ResMap algorithm was proposed [151]. ResMap detects the features of a model by fitting a 3D sinusoidal function in the different points of the volume, and saves the wavelength of the smallest sinusoid detectable above noise. Consequently, the ResMap algorithm produces a local resolution map and associates a distribution of values to the resolution of the density map.

2.4 Liquid-Phase Electron Microscopy

In the previous sections an overview on the state-of-the-art of EM has been provided, offering a wide and deep dissertation of the most important techniques and methodologies. Most importantly, the severe limitations of EM have been presented, highlighting how all the analysis run so far are conventionally performed on samples in solid state. However, imaging samples in liquid state was made possible by the advent of the LP EM, a technique that paved the way to the birth of a plethora of new research areas. In order to stress the importance of this technology it is worth mentioning that many samples analysed via EM behave differently depending on whether they are imaged in solid or in liquid state. It is well known that some types of particles imaged via cryo-EM exhibit preferential orientations, thus hindering the 3D data collections [152]. This phenomenon is due to the presence of hydrophobic patches on the surface of the particles, which bind to the air-surface interface, and hence lock particles in a specific position. Although such interaction does not cause structural damages, it has been proven that particles absorbed by the air-water interaction may become denatured at a fast rate [152]. Moreover, biological structures as cell, proteins, and membrane enclosed structure contain water, altering and sometimes requiring specific preparation techniques to be imaged via conventional EM [153]. Unfortunately, these techniques have the severe drawback of altering and sometimes damaging the sample, and introducing artifacts in the imaging process, as discussed in Section 2.2. Therefore, the possibility of imaging specimens via LP EM has the great advantage of preserving the structure hydrated in their liquid native state. Furthermore,
LP EM withstands a much higher electron dose than cryo-EM before the beam damage becomes visible. This difference in critical dose is related to the easier diffusion of reactive species away from irradiated region in liquid than in ice. An additional explanation may rely on the possibility that charge pairs caused by the irradiation recombine easily in water \cite{154}-\cite{155}. Most importantly LP EM enables the observation of the motion of the particles or objects dispersed in the media and their evolution in time. Observing the dynamical processes over time is of great importance in order to fully understand material properties and reactions that require liquid conditions.

The history of LP EM covers almost a century, as first attempts to perform imaging of wet samples can be dated in the early 40’s \cite{156}. In 1942 Ruska proposed a novel technology based on an open cell environmental TEM (ETEM) \cite{157}, while two years later Abrams and McBain prosed a closed cell technology \cite{158}. The former approach consists in using an environmental chamber in the sample area of a TEM, which is connected to the column of the microscope by two small orifices. The sample chamber is kept at a fixed relatively high value of pressure to create a wet area of the sample containing liquid and vapour. These fluids leaking out of the sample through the first small orifice are rapidly expelled by a pump that keeps this intermediate area at low pressure. This pump is hence called differential pump \cite{159}. Conversely, the closed-cell approach adopts a sealed chamber entrapping a thin layer of liquid. This technique requires the cell to be manufactured with electron-transparent materials, i.e. materials that do not interact with the electrons of the beam, in order to avoid undesired interactions. However, both open-cell ETEM and closed-cell technologies involved strong limitations: the open-cell ETEM presented many problems in achieving the proper values of pressure, consequently not being able to avoid evaporation of the liquid; whilst the closed-cell faced problems in controlling the thickness of the contained liquid, and in manufacturing cells
that do not interact with the electron beam. These limitations were enhanced by the technologies available at the time that slowed down the growth and the performances of EM techniques to image samples in liquid state. However, both approaches set the basis for modern LP EM technology [153].

In the following decades, new materials, the development of device and manufacturing techniques facilitated and boosted the advancement of LP EM. Firstly, the development of open-cell technologies allowed imaging not just specimens in liquid state, but also in thick layers of liquids [160]. Secondly, the development of the so-called liquid cell (i.e. an hermetically sealed enclosure), and the adoption of electron transparent and light materials, such as carbon foils, graphene sheets or SiN$_2$ microchips were pivotal for improving the closed-cell technology [10]. A careful design of the liquid cell may allow the use of flowing liquids [161]-[162]. The first version of the liquid cell for TEM was proposed only in 2010 and was implemented as a sandwich structure of two microchips, encasing the liquid on the inside, withstanding the vacuum and presenting in the middle a thinner area realised in SiN$_2$ - called window -, which lets the electron beam pass through and hit the sample [14]. It is important to mention that the windows bulges outward towards the vacuum, due to the differential pressure between the liquid cell and the vacuum in the column (Figure 2.19) [163]. Consequently, the thickness of the liquid increases, escalating the number of scattering events between the electrons of the beam and the media. Increasing the thickness of the liquid may severely limit the spatial resolution of the recorded images [163]. Nevertheless, this technology has been widely used in many different fields, studying electrochemical reactions [11]-[12], labelled biostructures in biological cells [164], nanocrystal growth [165], and tomography reconstructions of particles in liquid state [166], dynamic processes [163], which is analysed in details in Section 2.4.2.

In particular, Park et al. analysed the effect that the solvent evaporation under the electron beam has on a solution of platinum nanoparticles, characterised by high electron scattering strength, and can thus guarantee high contrast in the imaging process [165]. This study was permitted by in-situ LP EM, the only method producing continuous observations of nanoparticles in their liquid native state. Specifically, the images were captured with a camera acquisition rate of 30fps. In such images, the evaporation appeared as a change of contrast proportional to the observation time, with thin and thick areas appearing in light and dark contrast, respectively. Having access to high acquisition
rate camera enabled the reconstruction of the trajectories of the nanoparticles with high precision. Consequently, it was possible to highlight different particles behaviour, and hence a connection between the solvent evaporation and the motion of the nanoparticles in solution, which resulted in the formation of ordered arrays (Figure 2.20).

Figure 2.20: A schematic representation of the results obtained by Park et al. while investigating the effect of the evaporation of the solvent under the electron beam on nanoparticles in solution. The evaporation increases proportionally to time, driving the nanoparticles assembly. Image taken from the original paper [165].

Hermannsdoerfer and De Jonge exploited LP EM to investigate the dynamics processes of gold nanoparticles in solution, such as dissolution processes [163]. The acquisition rate of the camera was set up to 1.75s (i.e. about 0.57fps), and the resulting image sequence is depicted in Figure 2.21.

Figure 2.21: a-d) Sample images extracted from the LP EM series at 30s. The sequence shows the dissolution of gold nanoparticles in solution. Image taken from the original paper [163].
2.4.1 Brownian motion

One of the most evident phenomena that can be observed and studied via LP EM is Brownian motion. Brownian motion is a stochastic motion that modifies the displacement and the orientation of particles at the nanoscale, when they are dispersed in a liquid [167], due to collisions with the fluid molecules [168]. This motion was first observed in the first decades of the nineteenth century by Robert Brown, while analysing pollen in water, and then modelled by Einstein a century later [169].

Brownian motion is due to the continuous collision of colloidal particles with solvent molecules, causing their re-orientation in liquid. The result is a random motion of the particles without any particular direction and this movement is called diffusion. From a mathematical point of view, the Brownian motion can be considered as the result of the combination of two different components: translation and rotation. The magnitude of these two components is expressed by the translational diffusion coefficient \( D_t \) and the rotational diffusion coefficient \( D_r \), which depend on the same parameters, but are modelled independently. The equations expressing the translational and rotational diffusion coefficients of the Brownian motion in 2D are reported in Equation 2.8 and Equation 2.9, respectively [168].

\[
D_t = \frac{k_B T}{6 \pi \eta R} \quad (2.8)
\]

\[
D_r = \frac{k_B T}{8 \pi \eta R^3} \quad (2.9)
\]

where \( R \) is the particle radius, \( T \) is the temperature, \( k_B \) the Boltzmann constant and \( \eta \) the fluid viscosity, i.e. the resistance to the flown. It is important to stress the dependency of the motion to three main factors: the particle radius, the fluid viscosity and the temperature. Therefore, it is possible to control the diffusivity of the particles in liquid by modifying temperature and viscosity of the fluid. The Brownian motion can only be observed below the microscale, where the role of the fluid becomes of great importance, since low Reynolds numbers are implied. Reynolds number \( (Re) \) is a dimensionless parameter modelling the ratio between inertial and viscous forces within a fluid (Equation 2.10).

\[
Re = \frac{\text{Inertial Forces}}{\text{Viscous Forces}} = \frac{\rho v L}{\eta} \quad (2.10)
\]
where \( \rho \) is the density of the fluid, \( \eta \) the fluid viscosity, \( v \) the flow velocity and \( L \) the length of the object. When the objects are particularly small (low values of \( v \) and \( L \)) the Reynolds number is really low (\( Re \) much lower than one) and viscous forces dominate. This means that water behaves like a viscous liquid and hinders the objects in motion. Conversely, high values of \( L \) would make the effect of viscous forces negligible.

### 2.4.2 Nanoparticle imaging

In 2015 for the first time, a novel method to reconstruct the 3D structures of individual nanoparticles in solution was proposed [170]. This technique merged the technologically advanced graphene liquid cell, LP EM, a DDD and a computer vision algorithm for \textit{ab initio} 3D reconstruction. This study was able to recover the structure of platinum nanocrystals, exploiting their high electron density, a peculiarity that translates into high contrast of the nanocrystals when imaged via LP EM. In addition, the chemical resistance (\textit{i.e.} the ability of a material to maintain the original properties after being exposed to a particular chemical phenomenon [171]) of the platinum bestowed to the particles allowed for long time exposure to the electron beam. The sample in liquid state was encased in two graphene sheets, which guarantee a thin covering of material, hence protecting the liquid sample from the vacuum needed for EM imaging, and mitigate the beam damage effect [172]. This latter effect is due to ability of the graphene to absorb the radicals of media generated by the interaction with the electron beam, which

---

\[ \text{Figure 2.22: a) Sample frame extracted from the image sequence obtained via LP EM and depicting platinum nanoparticles in solution. b) 3D reconstruction of one platinum nanoparticle reconstructed from the image sequence in a). Images taken from the original paper [166].} \]
otherwise would interact with the specimen [173]. Thus, thousands of images depicting particles of the same species recorded at 50fps (Figure 2.22a) – or multiple samples prepared under identical conditions – showing different orientations were used to reconstruct the 3D volume of the nanocrystal (Figure 2.22b). In order to construct the output model, a specific algorithm named PRIME was tailored on the specifications of the micrographs [170]. Conventionally, 3D reconstruction algorithms average consecutive frames in order to increase the SNR. However, the particles to image in liquid state are free to move, thus resulting in different positions when captured in every frame. Hence, averaging consecutive frames creates a blurred, low quality image, which is not adapt for reconstruction purposes. In order to overcome this limitation, PRIME performs weighted orientation assignment and stochastic optimisation, without the need of any a priori information.

Initially, a reference model is created by randomly assigning angular orientations to the particles profile. This rough model is then iteratively refined by stochastic optimization of the correlation of the profiles with the reprojections of the reference model. At every iteration, the algorithm assigns a weight to every profile and then generates a weighted 3D volume. A continuous distribution of weights is then assigned to every profile, proportional to the correlation value (Figure 2.23).

![Figure 2.23: Schematic representation of the working principle at the base of the PRIME reconstruction algorithm. At every iteration of the refinement process a weight is computed and assigned to every profile (on the left) and a weighted 3D model is created. Thus every profile results to be determined by a continuous distribution of weights, proportional to the correlation of the model and the reprojections (on the right). Image taken from the original paper [174]](image)

The optimisation process is based on the first-improvement heuristic, interrupting the iteration as soon as the first new orientation that improves the previous correlation is found. This procedure guarantees the convergence of the optimisation problem to a local optimum, sacrificing the search of the best solution in favour of diversification. This ap-
approach forces the procedure to evaluate many more feasible orientations and combinations of orientations without being incredibly computational time-demanding.

Unfortunately, this reconstruction method was not applied to further studies, because of the strong limitations associated to LP EM and the sparse distribution of LP EM facilities all over the world.

2.5 Computer Vision

In the last decades, technology has continuously evolved reaching higher and higher peaks of complexity, with the aim to develop artificial systems that can imitate and replicate the human behaviour. In 1956 the research field known as Artificial Intelligence (AI) was created at a workshop at Dartmouth College by John McCarthy [175]. However, almost thirty years had to pass to see the field of AI blossom and hence receive important funds. This acceleration put an end to the dark era for the AI known as “AI winter”. This turning point was due to the tremendous development on electronics technology and manufacturing, which allowed the production of powerful devices at low costs [176]. From that point on, AI has flourished, pushing farther and farther the limits of many different research fields, such as planning, natural language processing, learning and robotics, amongst many [177]. One of the largest field of AI, which will be discussed in detail in this thesis, is Computer Vision (CV). This branch of AI comprises all the techniques that try to recreate the human process of extracting information from sight. However, humans recover information from sight by putting together knowledge about edges, shapes and volumes, colours, and depth. All concepts easy for a person to understand, but hard to translate into machine language. Therefore, CV can be described as an inverse problem [178].

Independently from the technique adopted to tackle CV problems, it is worth defining the common ground of this research field. Every method proposed to extract information out of images has its pillar on (one or more) input images. An image is a 2D projection of a scene captured by a detector, which is digitally modelled as a matrix. Every cell of the matrix is called pixel and contains a numerical value representing the intensity of the colour at that specific point of the image. The numerical values depend on the format chosen to express the values, but conventionally, the values span from 0 (minimum brightness) up to 1 (maximum brightness). If the image is black and white, the matrix is 2D and
the minimum value corresponds to black, while the maximum value to white. Conversely, colours in images are conventionally generated by adding different intensities of the three fundamental colours, red, green and blue (RGB). A machine implements coloured images as a 3D matrix (*i.e.* a stack of 2D matrix), with each layer associated to a colour [178]. A representation of the digitalisation of an image is depicted in (Figure 2.24). The choice of the image to use in Figure 2.24 is not random, as the portrait of the model Lenna Sjööblom has been adopted as standard to test CV methods since early 1970s [179].

![Figure 2.24: Schematic representation of the digitalisation of an image. On the left, a coloured image modelled as RGB, having a stack of three 2D matrix expressing the colours as combination of the pixels value. On the right, the same image, but expressed in black and white, thus having the shades of grey formed as the intensity values saved in a single 2D matrix.](image)

2.5.1 **Traditional programming versus Machine Learning**

During the last 50 years hundreds of methods were developed to process images, however a strict classification is difficult to realise, as different subfields may be created depending on the adopted classification *criterion*. Nevertheless, in this subsection a binary classification based on the philosophy at the base of the CV methods is proposed, distinguishing traditional programming from machine learning (ML). The former category heavily relies on the skills of the software developer, as traditional algorithms are defined by a series of strict rules to follow. Thus, software developers have to finely code the different tasks, taking into account all possible scenarios. Consequently, the higher the complexity of the tasks to perform, the less general purpose the algorithms to program. Conversely, ML-based approaches tries to derive information out of the data. Thus, the perspective is flipped, switching from tailoring a set of the rules on the data, to analysing the data to derive the ruling model (Figure 2.25) [180].
2.5.2 Machine Learning

In this thesis, methods belonging to both traditional programming and ML categories will be proposed, finely describing the assumptions and the rules implemented in case of traditional algorithms. Conversely, in this subsection an overview of ML and the logic at its basis is provided.

ML is a subset of AI that comprises a series of techniques that try to have a machine learning information from data, emulating the human learning process, as the name may suggest. Consequently, computers develop a learning technique without the need of human experience, teaching themselves how to behave in case of unseen data. Thus, ML-based approaches well fit complex problems, and adapt to unmodelled difficulties. ML-based approaches have become widely diffused since the early 1990s [180]. One of the first and most popular tool is the spam filter, which is a good example to describe the working principle of ML. Spam filters separate and hide undesired email, based on a list of examples named training set. Every item in the training set is named training instance. This method updates its training set every time a new spam email is marked by the user. In this fashion, the spam filter evolves and reacts to new scam methods. Traditionally, developing a similar algorithm would require an impressive amount of conditions and rules hard to define and keep updated [181]. Moreover, ML overcomes traditional programming limitations and is then adopted in many different fields, as representing high-dimensional data [182]-[183], face recognition [184], image classification [185], having applications responding to human voice [186]-[187], autonomous driving [188] and many others.

In the last decades many types of ML systems have been proposed that can be classified in broad groups based on human supervision, the ability to process online new data, or the implementation of the training process [181].
2.5.2.1 Human supervision classification criterion

The first category is structured on the amount of human supervision ML methods require during the training phase. There are four main categories:

- **Supervised learning**: the training set to feed to the ML approach contains both the training instances and the desired solutions, which are called labels. The main fields of application requiring supervised learning are classification (e.g. the spam filter) [189] and target prediction based on a set of features, a task also known as regression [190]. The most common supervised learning techniques are support vector machines (SVM) [191], decision trees and random forests [192], and artificial neural networks (ANN) [193].

- **Unsupervised learning**: the training set is unlabelled and the ML algorithm tries to find hidden patterns in the data. It is mainly used for clustering, generally implemented via K-means [194]; dimensionality reduction [195], including methodology as principal component analysis (PCA) [196]; and anomaly detection [197], implemented via SVM.

- **Semisupervised learning**: most of ML methods belonging to this category are a combination of the previous two, as they deal with partially labelled data.

- **Reinforcement learning**: the learning system (in this context called agent) analyses the environment and performs actions. According to the effects the action has on the environment, the agent receives a reward (or a penalty in case of negative rewards) that shape and influence the selection of the following actions, determining the best strategy (also known as policy) [198]. For example, this approach has been adopted to have a robot learn how to walk [199]-[200], or to develop AI able to play games. In this context the AlphaGO bot designed by DeepMind is very famous, as in 2017 it beat the world champion of GO [201].

2.5.2.2 Generalisation classification criterion

A second important classification is based on the method ML uses to generalise and then derive the learning model. This category is comprised of two main groups:
• **Instance-based learning:** the system memorises the examples and tries to generalise to new cases, based on the similarity measured between examples and new cases [202].

• **Model-based learning:** the system analyses a set of examples and makes a model out of them. This model is then applied to make predictions on new data [203].

### 2.5.2.3 Learning classification criterion

Another classification worth mentioning to define ML background concerns the ability of the systems to learn dynamically from a flow of incoming data [204]. Two main groups can be listed in this classification:

• **Batch learning:** also known as offline learning, is used to describe systems that can only train on the available set of data. In case of new data has to be included, it is necessary to repeat the training process, generating a new model that replaces the previous one.

• **Online learning:** the ML system is incrementally fed with examples organised in small groups, called mini-batches. In this fashion, the system can learn about new data on the fly, *i.e.* as they arrive. This learning procedure is successfully applied to systems that receive a continuous stream of data, or in case the training set is so big that it does not fit in the computer memory at once.

A fundamental parameter affecting the performances of online learning ML systems is the learning rate, a measure of the speed the system adapts to new data. High values of the learning rate cause the systems to quickly adapt to new data, but they also cause the system to forget old data. Conversely, low learning rate values make systems react slowly, but make it robust to noise and new unseen data to process [205].

In the previous sections the importance of data in machine learning has been highlighted, specifying how a sufficiently big dataset can replace a strict set of rules. In this way, the data chosen to create the training set heavily affect the goodness of a ML system and its results. The golden rule of ML is synthesised in “garbage in, garbage out”. If a training set contains errors and outliers (*i.e.* data significantly differing from other measurement), or data are heavily corrupted by noise, it is very likely that the system
will underperform. In addition, if the training set is nonrepresentative of all the cases to generalise to, the system may incur in prediction errors. A training set too small or too big may lead to sampling noise, and sampling bias, respectively. The former scenario produces nonrepresentative results in output; the latter occurs if the training set is biased towards specific subsets of data. However, the only method to understand if a ML system is incurring in some errors is applying it to new unseen data. The most common solution is splitting the dataset into training set and test set. Conventionally, the test set contains between the 20% and the 30% of the data composing the training set. As the names suggest, the former is used to train the system in order to generate a model, that is further tested on the test set. The measurement of the accuracy of the performance on the test set is named generalisation error [206]. The scenario in which a system performs well on the training set, but poorly on the test set is named overfitting. This problem may happen when a model is too complex or corresponds so closely to the training data, that is not able to generalise well. Moreover, the opposite scenario is named underfitting and occurs when the model is too simple to learn the structure modelling the data [207]. A schematic representation of the under/overfitting data is depicted in (Figure 2.26).

2.5.3 Artificial neural network

After introducing the working principles for ML, the present section aims to discuss the most powerful approach for implementing ML to solve complex problems: the ANN, or
simply neural network (NN). ANNs are nets of artificial neurons \(i.e\). the nodes) with the ability to model the behaviour of the human brain. In a biological neuron, the dendrites enable the cell to receive signals in form of electrical pulses from hundreds of surrounding neurons, which are then mixed together by the cell. After the electric potential inside the cell reaches a certain threshold, the cumulative signal is transmitted along the axon and fired in output. Similarly, artificial neurons are multi-input single output systems that weight \(\omega_i\) the different inputs \(x_i\), linearly combine them, bias the summation with a bias term \(b\) and set them in output \(y\) after applying the so-called activation function \(f\) (also known as firing function) (Figure 2.27) [180].

Figure 2.27: Schematic representation showing similarities between a biological and an artificial neuron. 

\(a\) The biological neuron receives the external electrical stimuli through the dendrites, which are further mixed by the cell. The signal is fired in output after travelling through the axon. 

\(b\) An artificial neuron transmit an input signal to output after performing a series of operations. The input \(x_i\) are weighted \(\omega_i\) and summed together with a bias term \(b\). Then the result is filtered by an activation function \(f\) and set in output \(y\).

The neuron structure described in Figure 2.27 is also known as perceptron, and from a mathematical point of view can be expressed as reported in Equation 2.11.

\[
y = f\left(\sum_{i=1}^{N} x_i \cdot \omega_i + b\right)
\]  

(2.11)

The weights are of great importance as they control the connections between neurons, selecting the relevance of each of them. However, the output of the neurons are controlled by the activation function, which acts like a switch and maps the output to a range of values. The most common activation functions are the heaviside step function, the Sigmoid, the rectified linear unit (ReLU) and the leaky ReLU (Figure 2.28). However, the
selection of which activation function relies on the mathematical properties each of them can assume [208].

![Figure 2.28: A series of the most common activation function controlling the output of the neurons is provided. From left to right a heaviside step function, a sigmoid function, a ReLu and the leaky ReLU.](image)

Heaviside and Sigmoid functions have the great advantage of normalising the output of the neurons to a range of values, conventionally between 0 and 1. This procedure makes the training method generally stable, as it avoids signal to diverge while transferring across the NN. However, these functions saturate high and low values, which results in low sensitivity. Consequently, the learning process finds difficult to adapt the weights to improve the performances if the output of neurons saturates. Conversely, both ReLU and leaky ReLU offer the great advantage of avoiding easy saturation of the neurons, and they result to be mathematically easy to implement. Moreover, the leaky ReLU introduces a small slope for negative input values, with the purpose of increasing the operational range of the neurons. However, the output values are not constrained into a range and both the ReLU and the leaky ReLU are non-differentiable (i.e. the tangent line passing through a point of the function is vertical) at zero, which may represent a problem during the training phase [209]-[210].

Once the structure of the neurons has been defined, it is possible to move forward analysing how these neurons connect creating a NN. Neurons are organised in layers, weighting the input data and transferring them in output. The series of input data is named *input layer*, the data in output is named *output layer*, while the layer of neurons in between is named *hidden layer* (Figure 2.29a). Thus, an NN is a simple structure, made of only three layers. However, the performances that can be produced by this system are
Increasing the number of interconnected hidden layers boosts the performances of the ML system, which is then named deep learning (DL) (Figure 2.29).

Figure 2.29: Schematic representation of the architecture of ANN and DL. a) An ANN is comprised of an input layer (in yellow), a single hidden layer (in green) and an output layer (in blue). b) A DL is defined as an ANN comprising more than a single hidden layer. This dense structure of the DL architecture highly increases the performances of the ANN.

The training is the core of the learning process. It relies on an iterative update of the values of the weights assigned to each neuron. The number of iterations during which a NN is training is called epochs. The number of epochs is a parameter the software developer decides before starting the training and it affects the performances of the NN. This refinement process is driven by a cost function (or loss function), the mathematical formulation of the problem to solve. In case of supervised learning, when data are labelled, the cost function consists of a minimisation problem of the error (also known as loss), i.e. a minimisation of the differences between the predicted and the actual values. In every iteration, the weights are computed, and back-propagated to adjust their value compensating the error. This process is also know as backpropagation [211]. Updating the weights in the backpropagation may be implemented by different techniques, the most popular of these techniques is the stochastic gradient descent (SGD). This optimisation algorithm computes the first order derivative (i.e. the slope) of the cost function with respect to every parameter in the network, thus computing the gradient. At this point, the values of the weights guaranteeing the steepest slope are saved and set in input to the next iteration. This procedure is iterated until the cost function value drops below a fixed threshold or for the total number of epochs, which condition comes first. It is thus evident
how the choice of non-differentiable activation function (\textit{i.e.} ReLU and Leaky ReLU) may represent a problem.

Designing and tuning an ANN to deliver enhanced performances for a specific field is a complicated and time consuming process. However, it is possible to adapt an existing ANN to a new and different goal, by recurring to the so-called \textit{transfer learning} technique [212]. This approach consists in using the architecture of an existing and pre-trained NN as starting point to design a new method. The weights produced after training the old network can be set as initial values to the weights of the new network, while the last layers can be replaced with layers best fitting the new needs. In this fashion, the new method borrows the ability to correctly extract useful information from the data, hence applying them to similar new tasks.

\subsection{2.5.4 Convolutional neural network}

In the last few decades, DL has overtaken traditional algorithms in the majority of research fields, due to the flexibility and great performances DL can guarantee on highly complex tasks. One of the field where DL delivers the best results is CV. As discussed at the beginning of Section 2.5, the design of traditional algorithms with the ability to extract data out of images is an arduous task. However, a specific class of DL was proven to produce excellent results in the field of CV. This class of DL is named \textit{convolutional neural network}, also known as CNN or ConvNets. In this thesis, only the expression “CNN” will be adopted [181].

CNNs have their root in a series of experiments on cats and monkeys in the late 1950s, which proved that neurons in the visual cortex of the animals react differently: different neurons respond only to a limited set of visual stimuli field, and detect only specific features (such as horizontal lines) [213]-[214]. Hence, the information recovered from all the neurons create a complete map of the visual field. Based on these biological studies, in 1980 Fukushima proposed the \textit{neocognitron}, the ancestor of modern CNN [215]. A milestone of CNN history was developed in 1998 by LeCun, who proposed two novel building blocks: \textit{convolutional layer} and \textit{pooling layer} [216]. Conventionally, CNN hidden layers are comprised of repetitions of several convolutional layers, followed by ReLU activation functions and by a pooling layer. Repetitions of these stack of layers constitute the main
structure of CNNs. The terminal layers of CNNs are generally represented by a few fully connected layers (FC) Figure 2.30.

![Figure 2.30: Schematic representation of the architecture of CNN. The input is processed by a stack of 2D filters via a set of convolutional operations. The results are then reduced by applying a pooling technique. This scheme is applied different times and the final output is then flattened by a sequence of fully connected layers.](image)

Each of these layers performs a very specific task and has its own architecture [217]:

- **Convolutional layer**: this type of layer is comprised of a stack of 2D filters, used to extract specific features out of the input volume, by convolution. This operation is the result of the dot product between the filter and a small region of the input, the outcome of which can be considered as a neuron of the layer. The filter scans the whole input volume, extracting features out of the images, and producing the so-called *feature map*. The first convolutional layers extract low-level features, as edges and colours. However, the deeper the layer, the more abstract the features to extract, since the network start looking for more detailed and exclusive features [218]. The stack of feature maps constitutes the output volume of the layer. During the training phase the weights of the filters are updated (Figure 2.31a).

- **Pooling layer**: this layer is used to subsample the input volume, as they reduce the size of the feature maps, while preserving the information. The neurons of the preceding layer are shrank and transmitted to the next layer. This operation can be performed by transmitting the highest (*i.e.* max pooling) or the average (*i.e.* average pooling) value of the neuron (Figure 2.31b).

- **Fully connected layer**: this layer connects every neuron of the preceding layer to every neuron of the next layer. In this way the stack of feature maps are flattened and transformed into 1D matrix (Figure 2.31c).
Figure 2.31: Schematic representation of the working principles of most used layers in CNNs. a) In convolutional layers the input is processed via a series of 2D filters that extract information and generate feature maps. b) Pooling layers reduce the size of the input, while preserving the information. The pooling operation can be performed in two different ways: extracting the maximum value out of a region of pixels (max pooling), or computing its average (average pooling). c) Fully connected layers connect all the neurons of the preceding layer, with all the neurons of the next layer, flattening the output.

CNNs are amongst the most popular DL networks, as they produce great results in many fields, such as face detection [219]-[220], object detection [221], image classification [222], or image segmentation [223].

2.6 Computer Vision Applied to Electron Microscopy

In the previous sections an overview of EM, its strong dependency to the images and the technological field dealing with image processing have been provided. In this section, a connection amongst these three elements is highlighted.

In Section 2.3 the importance of algorithms processing images to recover information on the structure of samples and thus their functionality has been highlighted. However, the applications of computer vision to EM is not limited only to 3D reconstructions, but cover many fields, as image enhancement, pre-processing, and particle detection, amongst many.

2.6.1 Image enhancement

The majority of methods belonging to this category aims at increasing the SNR of images, while preserving relevant details. From a mathematical point of view, an image corrupted by noise can be modelled as reported in Equation 2.12 [224].

\[ I_N = I_O + N \]  \hspace{1cm} (2.12)
where $I_N$, in this case, models the images recorded via EM and corrupted by noise, which are the result of the the sum of the actual noiseless image ($I_O$) plus the noise function ($N$), here considered as additive. Simple algorithms of noise reduction usually process images by applying a filter, which may cut out some components of the original signal. One of the simplest and most popular adopted filter is the Gaussian filter, which is modelled as a 2D Gaussian distribution. This filter is scanned over the noisy image and smooths all the details down, without distinguishing between signal and noise. Consequently, this approach has the downside of removing important features of the image [225]. The intensity of the blur is proportional to the size of the filter, as seen in Figure 2.32. However, Gaussian filter owes its popularity to the low processing time required to enhance images.

![Original, Noisy, Gauss 5px, Gauss 7px, Gauss 11px](image)

*Figure 2.32: Schematic representation of the working principles of a Gaussian filter. The original image (a) is corrupted by noise (b) to verify the goodness of the filtering process. Applying a Gaussian filter has the great advantage to smooth noise down. However the size in pixel of the filter blurs the image, losing details. Images c, d and e were obtained by applying filters wide 5, 7 and 11 pixels, respectively. All the images are made of 512 x 512 pixels.*

In order to avoid the smoothing limitation, complex denoising approaches have been proposed. The *bilateral denoising filter* is widely used in EM, and consists in a non-linear filter resulting from the combination of two low-pass filters (*i.e.* a type of filter cutting out high frequency information). The first filter is a classic low-pass filter that cuts out high-frequency components (*i.e.* small details) proportionally to the distance from the center of the filter. Conversely, the second filter weights its components proportionally to the difference of intensity with respect to the value of its central pixel. Therefore, the higher the value of the difference, the smaller the contribution of the pixel [226]. In this fashion, the bilateral denoising filter reduces the noise component, without compromising the resolution.

The impossibility to produce noiseless images puts an obstacle to the diffusion of DL-based approaches, as the majority of denoising systems developed via CNNs are based
on the analysis of noisy-clean pair of images [227]-[228]. However, in the last few years a novel denoising method for EM images based on CNN was proposed, without any need of noiseless images [229]. The proposed approach tries to predict the noise corrupting every pixel, and requiring only noisy images in the training set. Building on a similar architecture, other CNN-based methods were proposed, reaching great results in denoising EM images [230]-[231]. In addition, a specific type of CNN named autoencoder had been proven to produce interesting results in the denoising field. This architecture is comprised of two CNNs, and is used to extract features out of a set of data in unsupervised manner. The first part - named encoder - extracts the features out of the input, and reduced the size of the input layer by layer. The last layer of the encoder is named bottleneck, and from that layer on the decoder starts reconstructing the input data from the extracted features Figure 2.33.

![Figure 2.33: The DAE is comprised of two CNNs, the first one named “encoder” and extracts the features out of the input image. The second CNN named “decoder” reconstructed the noiseless image starting from the previously extracted features. The contact layer between the two CNNs is named “bottleneck”.

The assumption at the base of a denoising autoencoder (DAE) relies in the property of noise to be incompressible. Consequently, the encoder is able only to compress image details, neglecting the noise corrupting the image. In this fashion, the decoder processes only the image features, reconstructing a noiseless image [232]. The DAE has been presented as a method that reconstructs the noiseless images by starting from their noisy versions. However, by design DAEs require two different inputs to estimate the noise distribution. This constraint does not easily appease with the absence of ground truth. Therefore, a copy of the noisy images is slightly corrupted and set in input. This solution prevents the
DAE from learning the so-called *identity function* in case two copies of the same input were selected, and allows the DAE to model the noise [233].

Conventionally, DAE can be trained by using two different models of cost functions: the mean-squared error (MSE) modelled in Equation 2.13, or the binary cross-entropy (BCE) in Equation 2.14.

\[
l(f(x)) = \frac{1}{2} \sum_k (\hat{x}_k - x_k)^2 \tag{2.13}
\]

\[
l(f(x)) = - \sum_k (x_k \log(\hat{x}_k) + (1 - x_k)(\log(1 - \hat{x}_k))) \tag{2.14}
\]

In both Equation 2.13 and Equation 2.14, \(x_k\) represents the \(k\)-th element of the input vector, while \(\hat{x}_k\) represents the \(k\)-th element of the output vector, which in case of a DAE is the additionally corrupted input transformed by the encoder first and the decoder then. The MSE function minimises the distance between input and output, while forcing the DAE to learn to remove the corruption. Conversely, the BCE maximises the value of \(\hat{x}_k\) during the training phase forcing the DAE to learn a better and better noise model [234]. However, in case the input values are in the range of \([0,1]\), the BCE was proven to produce better results than MSE [234].

### 2.6.2 Particle detection

In Subsection 2.3.2 different software developed to detect particles in EM tomographs for SPA are listed. The following section aims to discuss the implementation of identification algorithms, which may be grouped mainly into two categories:

- **Non-learning methods**: this category includes approaches based on template correlation [235]-[236]-[237], edge detection [238] or intensity level [235]-[239]. Template-based methods rely on the assumption that particles in micrographs can undergo small variations, appearing only in few different poses. Thus, creating a template to identify via correlation guarantees great results. Conversely, edge detection-based methods computes edges defining geometrical shapes and identifies those as particles [240]. On the other hand, intensity level methods exploit the contrast produced by particles in LP EM, which results dark on a bright background, when imaged in
bright field. Thus, algorithms based on the interpretation of the intensity level generally cluster together areas of the image containing low intensity pixels. However, all these approaches strongly depend on the intensity value of the pixels, they are thus sensitive to noise, and they are also computationally demanding.

- Learning-based methods: conventionally they are structured over two-stages, the first stage is learning, during which a model of particles is estimated from a dataset; the second step is particle detection. It is easy to understand that the algorithms belonging to this category are generally based on NNs [241]-[242]. Additionally, a subset of this class of methods worth mentioning include particles identification methods based on CNNs, such as \textit{cr-YOLO} [243].
CHAPTER 3

METHODOLOGY

In the previous chapter, an overview of the history of LP EM has been presented, highlighting the similarities and differences with respect to cryo-EM, and analysing the methodology currently dominating the state of the art in the fields of image analysis and 3D reconstructions. Despite many analogies, the software developed for cryo-EM do not account for the features of the images produced via LP EM. In particular, the presence of the liquid lowers the quality of the recorded data, by adding unprecedented noise and blur components, that thus require the development of novel techniques. Moreover, designing a new method to perform the reconstruction task is further complicated by the presence in the input data of a fourth dimension: time. Conventional techniques extract information from images of immobilised particles. Unfortunately, these methods result to be invalid when applied to particles that are able to freely translate and rotate, or even evolve in time. Conversely, in the present investigation time and dynamics are exploited, driving the development of novel methods to analyse these temporal phenomena. All the elements described in Chapter 2 will be fundamental to fully understand the computer vision (CV) methods proposed in this chapter.

3.1 The Equipment

First, in order to understand the peculiarities of this work, it is necessary to analyse in details the equipment used to produce the data to analyse and process in this work. All the images included in this thesis were recorded via LP EM, by means of a JEOL JEM-2200FS TEM microscope, equipped with a FEG at 200kV and an in-column Omega filter. This filter is named after the omega (Ω) shape formed by four additional electromagnets in the column and is used to acquire zero-
loss images, *i.e.* images created by electrons without energy loss [244]. The microscope is paired with a K2-IS camera from Gatan, an ultra-high sensitivity DDD detector covered with a fiber optic layer and a scintillator, in order to allow low-dose imaging, thus increasing the radiation resistance (*Figure 3.1*). Moreover, this detector can record images at very high spatial and temporal resolution, up to 3838x3710 pixels and 1600fps, respectively. The K2 detector can run in two different configurations, performing in different fashions:

- *In-situ (IS):* the detector transforms the incoming primary electrons into a digital image, which is further processed in real-time and stored. The processor can correct the images by compensating the PSF and bin them, in order to mitigate the noise. The K2 camera used in IS configuration can record up to 1600 frames per second.

- *Summit:* the electrons are transformed into digital images and transmitted into a processor, which can respond to every electron event as soon as it hits the detector. In this configuration, the camera outputs constantly 400 frames per second.

The Summit configuration enables the camera to capture images in three different modes, adapting to the needs of the user:

1. *Linear:* the detector allows the user to continuously observe the specimen, but records only a single frame at a time.

2. *Counted:* neighbourhoods of four pixels are combined into a single one, thus increasing the robustness of an image to noise (*Figure 3.2b*).

3. *Super-resolution:* the resolution of the image in output is digitally doubled (from 3838x3710 to 7676x7420 pixels), pushing the detection of the electron events to sub-pixel precision (*Figure 3.2c*).

This detector allows the user to record a series of continuous images, thus registering a video of the sample free to move in liquid in three dimensions. The most important element of these videos is the *frame rate.* It measures in frames per second the time interval separating the acquisition of two consecutive frames, and at the same time the exposure time of the detector. In our scenario, the higher the frame rate, the more refined the dynamics monitored via LP EM. However, the smaller the frame rate of the camera, the shorter the acquisition time, and the number of electrons hitting the detector. Recording
Figure 3.2: Schematic representation of the counted and super-resolution imaging modes of the K2 camera in Summit configuration. **a)** The electron (in black) hits the detector and its signal is scattered (in yellow) over different pixels that transform the energy in a digital value. **b)** The electron event is reduced to the highest charge on the pixels. **c)** The charges are localised with sub-pixel accuracy.

images at high frame rate has the downside of generating dark images with low contrast, when organic specimen are imaged.

In *Section 2.4* the complexity of LP EM imaging technique and the technological requirements have been described, highlighting the fundamental role of the liquid cell. In order to perform LP EM, the Ocean liquid holder manufactured by DENSsolution was employed. This holder is designed as a closed-cell made of two microchips of silicon nitride, which entrap the liquid on the inside, while avoid interferences with the beam, as its central area is electron transparent. These two microchips host at the center a 50nm-thick rectangular observation window, measuring $50 \times 200 \mu m$. Although the two microchips are generally separated by a 200nm spacer, the window experiences a bulging effect (*Figure 2.19*), due to the differential pressure between the liquid cell and the vacuum in the columns. This phenomenon leads to an increment of the thickness of the comprised volume of about 150nm. However, this side effect is less experienced at the corner of the observation window. Consequently, it is recommended that LP EM is performed close to these side areas of the window, thus facilitating good penetration of the electrons, and thus avoiding an additional undesired blur effect of the images.

In *Figure 3.3* a simplified geometry of the liquid holder reconstructed using COMSOL Multiphysics is proposed. The model has been digitally reproduced in detail following the actual dimensions of the holder. However, the physical dimensions of the design cannot be reported in this thesis in order not to violate the non disclosure agreement signed between the author and the manufacturing company.
Figure 3.3: Schematic representation of Ocean liquid holder manufactured by DENSsolution used to perform LP EM. On the left the bottom view of the holder, on the right a cross section. A flower-shaped tip hosts two microchips on the inside separated by a spacer that creates a microfluidic channel. The SiN microchips have a central area (in yellow) to let the electrons pass through (observation window). Two parallel in plane pipes are connected to the tip of the holder, allowing the fluid to flow in and out the liquid cell.

The tip of the holder is flower-shaped and connected to two parallel in plane pipes, an inlet and an outlet pipe for flowing the solution in and out, respectively. The tip of the holder encases on its inside two microchips separated by a variable spacer, which creates a microfluidic channel to keep the sample hydrated. This model has been used to perform fluid dynamics simulation to analyse the fluid flowing inside the holder, in order to understand the distribution of the particles. The simulation was performed at atmospheric pressure conditions, i.e. when the sample is flown into the liquid holder in the bench outside the microscope. Thus, the bulging effect of the window has been excluded from the simulation. The windows have been modelled as rigid, flat surfaces that entrap the liquid between bottom and top. These settings focused the fluid simulation on the flow distribution of the liquid, allowing an analysis of the sample distribution across the flower-shaped cell. In order to replicate the actual behaviour of the liquid cell and guarantee good fluid circulation, input and output ports were set at the end of the pipes, and the two microchips defining the closed-cell (Section 2.4) were initially removed from the holder. An incompressible Newtonian fluid was pumped in the empty chamber with an initial velocity of 5µL/min. The analysis was performed in steady state condition, imposing laminar flow, planar symmetry and no slip condition on the walls of the holder. The results of the simulation are discussed in Chapter 4.

The samples can be injected in the holder statically by drop casting a volume of the sample on to the windows, or by recurring to a peristaltic pump. In this latest approach,
the tubing system is connected to the liquid cell after the liquid holder is sealed. The sample in liquid solution is injected at constant speed until filling the liquid cell. The peristaltic pump is then stopped and the inlet and outlet tubes are sealed to ensure a closed liquid circuit. The micrographs are recorded five minutes after the cell is filled, in order to guarantee that convection effects from the flowing process were not affecting the Brownian movement of particles of ferritin protein in solution. Moreover, in order to improve the effect of the Brownian motion, the chips of the liquid cell are plasma cleaned, thus increasing the hydrophilicity.

The last piece of equipment worth mentioning is the workstation. This machine was used to run every software designed and presented in this thesis. It runs Ubuntu Linux 16.04 64-bit and has the following hardware specifications: two central processing unit (CPU) Intel Xeon Gold 5118 2.3GHz with 12 cores each, 128GB (8x16GB) 2666MHz DDR4 RAM, 512GB class 20 solid-state drive (SSD) and dual SLI NVIDIA Quadro P5000 16GB as graphic processing unit (GPU).

3.2 Noise Reduction

One of the first approaches evaluated to perform 3D reconstruction of particles imaged via LP EM was based on the scale-invariant feature transform (SIFT) algorithm [245]. The SIFT method was proposed by Lowe in 1999 and represents a milestone in the 3D reconstruction field. This method produces a collection of local features that are invariant to translation, rotation, illumination changes and scaling. Matching the features extracted from images of the same object showing different profiles is a relatively easy task and reduces the 3D reconstruction into a triangulation problem [246]. Many different 3D reconstruction algorithms were developed and reported [247]-[248]-[249]-[250]. Having a 3D reconstruction method based on the identification of local features invariant to rotation and scaling would have perfectly fitted the LP EM requirements. Particles in liquid can translate across the detector plane, but also along the perpendicular - conventionally Z - axis, thus appearing close or far from the detector across a sequence of consecutive images.

Unfortunately, the presence of noise, together with the presence of the liquid surrounding the sample significantly lower the quality of the micrographs. Thus, SIFT-based methods resulted impossible to apply to LP EM micrographs, as the resolution of the recorded images was too low to identify the key features. This consideration pulled the
trigger to the investigation in the denoising field, with the aim of recovering the original image lying under the noise, as discussed in Subsection 2.6.1. CV comprises a vast amount of different denoising algorithms [251], which have been tailored on a particular imaging technique and type of noise [252]. Therefore, a general high-performance algorithm that processes in all the existing imaging modalities has never been developed.

Furthermore, the data acquired with the instrumentation described in the previous section are in the form of one minute long videos. From a digital perspective, working with a series of consecutive images rather than a single image brings in the concept of redundancy. Consecutive images may be characterised by similar patches at different locations. Therefore, the majority of denoising algorithms for videos extracts similar information from different locations of different images to estimate the distribution of noise corrupting the sequence [253]-[254]. Unfortunately, this approach is not valid in the case of LP EM videos, due to the local-dependency of the liquid affecting the imaging conditions. Therefore, the method to develop has to deal with one image at a time and patch by patch.

### 3.2.1 End-to-end pipeline

This subsection describes the first method implemented to reduce the noise corrupting LP EM images and has been proposed by Marchello et al. [255]. This approach is based on a more complex model of a digital image than Equation 2.12, and is reported in Equation 3.1 [224].

$$I_N = I_O \ast B + N$$  \hspace{1cm} (3.1)

$B$ is the blurring term that is convoluted to the original image $I_O$, modelling the blur effect smoothing down the details of an image. The aim of this method is not only to suppress the noise ($N$), but also to identify the blurring function ($B$), in order to recover high-quality details. However, the noise corrupting EM images is caused by a high variety of different sources, such as fluctuation of the intensity of the beam, secondary electrons and readout noise of the detector [17]. Hence, estimating the noise distribution is hard [256]. In case of conventional TEM, the blurring function does not represent a problem, as it is overcome by compensating the CTF. Unfortunately, none of the existing image
restoration algorithms developed for conventional TEM account for the presence of liquid \cite{17}.

The algorithm proposed in this subsection is comprised of two main steps organised in a pipeline that receives in input the noisy data, and produces in output an enhanced version of them. Firstly, the algorithm tries to identify and suppress the noise function. Secondly, the blurring function is estimated, in order to restore the sharpness of the image. In Figure 3.4 the block scheme modelling the pipeline is depicted, highlighting the connections between the processing steps. Swapping the order of these tasks would have unpleasant consequences, since estimating the blur function before suppressing the noise may actually amplify the noise corrupting the images \cite{257}.

![Figure 3.4: Flow chart modelling the end-to-end denoising pipeline adopted to restore high-quality images recorded via LTEM. The noisy image $I_N$ is preprocessed to remove salt-and-pepper noise ($I_{Nd}$) and any source of bias. The output of this step is processed to remove noise ($I_B$) and then deblurred. This process requires $I_{Nd}$ and $I_B$ in input to produce high quality results.](image)

The flow chart modelling the end-to-end denoising pipeline shows four different processing blocks. The first two blocks define the “pre-processing” stage and aims to improving the quality of the micrographs without losing high-frequency details. The very first block receiving in input the noisy image $I_N$ deals with the so-called salt-and-pepper noise. This type of noise corrupts images by adding very black and very white pixels to the image. Fortunately, this noise component can be easily removed by means of a median filter \cite{258}. It replaces the value of each pixel with the median value of the neighbourhood around the pixel itself. The filter is then applied over the full image, processing a pixel at a time. Moreover, the output of this block ($I_{Nd}$) is set in input to the “bias removal” block, in order to compensate the uneven illumination. It may occur that the center of the beam is not centered over the imaged area in the field of view, thus resulting in a nonuniform illuminated image. As a consequence, regions of these images may behave differently under the same processing steps. Therefore, in order to balance this effect the
image $I_{Nd}$ is split into different partially overlapping areas, the mean values of which are further removed. Consequently, all the image patches result to have the same mean value, as they are centered in 0. This mean removal process ensures a uniformity of responses of all the patches to the same processing steps [259]. The implementation of this process depends only on two variables: the size of the patches, and the *stride*, a value measuring the distance in pixels between two consecutive overlapping patches. Different values for these parameters have been tested, leading to the following considerations. Large patches include much information, and the mean value removal may thus heavily alter the intensity of the pixels in the patch. Conversely, small patches may be comprised of similar pixels, the mean values of which may not significantly differ from the single values. Moreover, the choice of the stride affects both the processing time and the goodness of the result. A very small value produces a smooth effect on the output, finely processing the whole image, but causes the algorithm process a high number of patches. Unfortunately, the number of patches to process proportionally increases the processing time associated to a single image. High values of stride however leads to process few distant patches, resulting in visible differences of intensity.

At this point of the pipeline, the pre-processed image is fed in input to the *denoising* step. It is worth reminding that the choice of the denoising algorithm has to take into account the local properties of the image, together with the need to preserve the details. In order to perform this task, the progressive image denoising (PID) algorithm has been implemented [260]. The PID algorithm is comprised of different stages. Firstly, PID iteratively applies gradient descent to produce a robust estimation of the noise in both spatial and frequency domain. The noise component is subtracted at every iteration and the kernel modelling the noise distribution is updated. The so-estimated kernel is then used to denoise the input image by applying a dual-domain image denoising (DDID) algorithm [261]. This method combines a bilateral filter in spatial domain and a noise-reduction technique in Fourier domain named *wavelet shrinkage*. The PID method brings many advantages: it is simple to implement, it provides short processing time and does not produce any artifacts in the denoised images [260]. The final stage of the proposed pipeline aims at restoring the sharpness of the image, by estimating the $B$ term of Equation 3.1. The method to solve this problem has to take into account different values of thickness of the liquid, since the bulging phenomenon prevents the estimation of a global blurring
function, as discussed in Section 3.1. Thus, from a mathematical point of view this problem reduces to an estimation of local kernels modelling the blurring function, which are further deconvolved from the image to restore the sharpness. This scenario is described as blind deconvolution. The method developed in this study is a local implementation of Yuan deblurring method [262], and will be referred to as LED (local extension deblur).

Yuan method was first proposed to correct the motion due to low exposure time affecting hand-held cameras. It corrects the blur by processing in input not only the blurred image, but also the noisy images. The former contains low frequency information, but low noise. Conversely, the noisy image contains high frequency details, even if corrupted by the noise. Thus, this method recovers information by combining the two input images. This method builds on the assumption that the original image \( I_O \) in Equation 3.1 can be considered as comprised of a denoised image \( N_D \) and a residual image \( \Delta I \) containing the sharp structures (Equation 3.2).

\[
I_O = N_D + \Delta I \quad \text{(3.2)}
\]

Given that \( \Delta I \) contains high frequency information, it results to be much smaller in intensity than \( N_D \), which thus represents a good starting point to start estimating the \( B \) term of Equation 3.1. By combining this equation with Equation 3.2, the problem of estimating \( B \), given \( N_D \) and \( I_B \) becomes a non-blind deconvolution (Equation 3.3).

\[
I_N = I_O * B + N = (N_D + \Delta I) * B + N = I_B + \Delta B + N \quad \text{(3.3)}
\]

where \( I_B \) represents the blurred image set in input, while \( N_D \) is the noisy image set as second input. Once the blurring function has been modelled, it is possible to directly recover the original image \( I_O \) by deconvolution. However, this method may introduce artifacts. Generally, deconvolution causes copies of features appearing around sharp edges, also known as “ringing effect”. Conversely, applying the deconvolution to the residual blurred image \( \Delta B \) significantly reduces the possibility to have artifacts. Moreover, the method is iterated to refine the estimation of the blurring function and further suppress the artifacts.

The LED implementation locally applies Yuan method to small portions of the input images. This approach is further repeated over the whole image, independently treating
all the patches. Consequently, the local implementation tackles the different blur effect introduced by the variation in thickness of the liquid in the sample holder. Similarly to the global Yuan method, the LED algorithm requires in input two images. In this pipeline, the output of the PID block is selected in input as blurred low noise image $N_D$; while the raw input image processed by a median filter $I_{Nd}$ as the sharp noisy image. In this fashion, the salt-and-pepper noise is excluded from the process, without affecting the deblurring phase. Furthermore, the results of LED algorithm varies with respect to the values of patch size and stride, as discussed for the mean removal step. The implementation of this method is free to use and available in a GitHub repository at the address https://github.com/GabrieleMarchello/LPEM-post-processing-pipeline.

The results of the end-to-end denoising pipeline are discussed in Chapter 4. However, it is worth anticipating that this method may be unfit to process long sequence of images, as it takes a few hours to process a single micrograph. Consequently, a low processing time method became necessary to process the large amount of data produced via LP EM imaging.

3.2.2 Denoising autoencoder

In Subsection 2.6.1 the problems in producing effective DL denoising approaches have been discussed. Nevertheless, a type of DL named autoencoder has been proven to produce great results without the need of the ground truth. In this subsection, the implementation of a DAE is discussed. Initially, a simple approach with few layers has been proposed in order to test the goodness of this method. Then, the architecture has been enriched and made more and more complex in order to improve the achieved performances.

As discussed in Subsection 2.6.1, DAEs require by design two different inputs to estimate the noise distribution. Conventionally, the second input is chosen as a replica of the first input images additionally corrupted by low intensity noise, in order to prevent the DAE to learn the identity function. In this work, the additional noise has been modelled with Gaussian distribution, with mean and variance values of 0.5 and 0.1, respectively. Such values were chosen in order to have low magnitude noise, avoiding to cover the actual content of the images, but instead triggering the noise estimation.

Firstly, the skeleton of the DAE was designed using an encoder and a decoder made of five layers each, realised as a sequence of convolutional and max pooling layers. The
encoder is comprised of three convolutional layers followed by a ReLU activation function. The kernel size is set equal to five rows and five columns - (5,5) - for the first layer and (3,3) for the others. The number of filters in each layer starts at sixteen, but doubles up at every convolutional layer reaching a value of sixty-four in the last layer of the decoder. The stride is kept equal to one in both dimensions. Conversely, the stride in the max pooling layers is set equal to two in both dimensions, same values assigned to the pool size. Similarly, the decoder is constructed by using the same parameters, with the filters disposed in a descending order, and by replacing the pooling layers with upsampling layers. As the name suggests, this type of layers increases the size of a matrix, filling the new pixels by applying the bilinear method. Moreover, the decoder is followed by a further convolutional layer activated by a Sigmoid function and made of only a single filter. This output layer shapes the image in the desired output format. The goodness of this architecture has been quickly tested by using images four times smaller than the original micrographs.

However, reducing the size of the images implies losing details, which is the main point of this whole analysis. In order to process full size images, the network has been modified by replacing the ReLU activation function with the leaky ReLU in every convolutional layer and the batch normalisation has been adopted. The images in input are loaded in batches in order not to saturate the memory of the machine and their values are normalised in order to have the images respond uniformly. This technique was proven to make the network robust and fast [263].

Once the architecture has been defined to process full size images, the focus of the study moved to the quality of the results. The so-defined DAE produced low noise images, but unfortunately the results were not as good as expected, highlighting a significant lack of details. This problem may be due to the compression introduced by the pooling layers, and this problem is progressively worsened by the depth of the network. The higher the number of layers, the higher the amount of lost details. Therefore, a simple solution to overcome this problem consists in carrying image details from the encoder to the decoder, hence skipping many compression steps. This approach is known as skip connection [264]. In this work, skip connection is implemented symmetrically by connecting the output of the first and second convolutional layer to the input of the last and second last convolutional layers, respectively. Thus, the inputs to the last two convolutional layers result to be
a linear combination of the skip connected-signal and the output of the previous layer (Figure 3.5).

Figure 3.5: Schematic representation of the DAE with symmetric skip connections. The output of the first and second convolutional layer of the encoder (c1 and c2, respectively) is carried in input to the second last and last convolutional layer of the decoder (c5 and c6, respectively) in order to preserve the details. The blocks are colour-coded: convolutional layers in yellow, pooling layers in green and upsampling layers in blue.

Training a DL-based method is as important as the architecture of the DL itself. Choosing the right values of the training process and creating a proper dataset is fundamental to the good performances of this DAE. The training process minimises the binary cross-entropy (BCE) loss function, by adopting the stochastic gradient descent (SGD) as optimiser, with a learning rate of 0.001. The training process is iterated for 100 epochs, with a batch size of 6 images. The training set is comprised of 450 full size images, the 20% of which was used to define the test set. The images used to define the training set are extracted from 9 different videos of SiO$_2$ nanoparticles imaged via LP EM (Figure 3.6). Further information on the sample depicted in the images composing the training set will be provided in Chapter 4.

3.2.3 Noise2Void

At the time the DAE discussed in the previous subsection was developed, a similar autoencoder was published by NVIDIA - the technology company -, in collaboration with Massachusetts Institute of Technology (MIT) and Aalto University. This deep DAE, named Noise2Noise (N2N), was built on the assumption that it is possible to “learn to turn bad images into good images by only looking at bad images”[265]. Thus, N2N was developed in order to denoise images, when the corresponding ground truth was unavailable. It was initially designed to work on magnetic resonance imaging (MRI) data,
Figure 3.6: The database to train the DAE is performed with 450 different images extracted from 9 different videos. The mosaic is comprised of one instance extracted from each video of the dataset, showing nanoparticles in liquid state imaged via LP EM. The images have been pre-processed by applying a Gaussian filter and the contrast has been enhanced by 30\% for displaying purposes.

but was soon extended to photographs and other type of noisy images. Similarly to the DAE in Subsection 3.2.2, N2N requires in input two images of the same subject corrupted by different noise distributions. The training is performed by minimising the MSE between a great amount of pairs of images. Moreover, this method was proven to produce great results on different noise distribution, such as Gaussian, Poisson and Bernoulli.

N2N is an asymmetrical autoencoder with skip connections. The encoder is comprised of seven different convolutional layers, with forty-eight filters each. Every convolutional layer is followed by a max-pooling layer that halves the size of the input vector. Conversely,
the decoder is comprised of ten convolutional layers, with different number of filters. All the layers have ninety-six filters each, but the second last and last convolutional layer have sixty-four and thirty-two filters, respectively. The convolutional layers are organised in series of two and each series is preceded by an upsampling layer. The skip connections carry the output of every max-pooling layer to the output of the corresponding upsampling layer.

However, N2N was never implemented and applied in this work, as only a few months later it got published, a novel DL-based approach was proposed. It is named Noise2Void (N2V) as it denoises images without the noiseless counterpart, but requires only one single noisy image in input [266]. Consequently, N2V can be classified as a self-supervised denoising technique. The choice of the name highlights the connection and at the same time the differences with N2N. This DL-based approach is based on the image model proposed in Equation 2.12, but it adopts a probabilistic model of the image, which is expressed in Equation 3.4.

\[
p(x, n) = p(x)p(n|x) \tag{3.4}
\]

where \(p(x)\) models the distribution of the signal in the image, while \(p(n|x)\) is the conditional probability distribution of noise for a given signal. Furthermore, the pixels \(x_i\) of the signal are not statistically independent. Conversely, the noise corrupting the pixels \(n_i\) is considered independent. Building on these assumptions, the network estimates the noiseless value of a pixel, by receiving in input the neighbourhood of the pixel, without the pixel itself. This structure is named by the authors blind-spot network [266]. Moreover, the same neighbourhood comprising the pixel is set in input as noisy target. Removing a single pixel from one input prevents the network from learning the identity function.

In order to maximise the performance, the network does not train on full images, but randomly extracts squared patches of the same size. A small portion of each patch is then fed in input to the training process, after masking out the central pixel. In this fashion, the optimisation of the cost function becomes a minimisation problem between the patches with the masked pixel and the original patch (Figure 3.7).

This DL-based approach has been widely used in this thesis, becoming a standard preliminary step in the pre-process of LP EM images. Unfortunately, the variation of the imaging conditions with respect to the samples to analyse, together with the locally
dependency of the liquid thickness do not allow for the creation of a standard denoising model. Therefore, N2V has to be trained every time a new set of images is recorded. Consequently, the number of the images comprising the training set depends on the available data. However, the number and the size of patches to extract from each image is kept constant, following the instructions given by the author. Hence, in the case of images containing more than 1024x1024 pixels 128 different patches are extracted from each image, keeping their size constant at 64x64 pixels, independently from the size of the image. Moreover, the training process is generally run for 100 epochs over the 75%-80% of the extracted patches, while the rest is separated into the test set.

### 3.3 Particles Identification

One of the greatest advantages of performing a LTEM imaging consists in the ability to investigate particles in their pristine state and their evolution over time. Thus, it is possible to observe the motion of the particles in liquid, or record their temporal trajectories. For this purpose, it is of great importance to identify and distinguish all the particles recorded in a frame.

In this section three methods to identify particles in liquid media are discussed. The first two methods are traditional algorithms, designed to work for as many imaging conditions as possible. Such traditional algorithms are not based on any *a priori* assumption about the shape and the size of the particles. The second method is an upgrade of the first algorithm. Conversely, the third method named *Deep LiqID* is performed via DL.
Additionally, in Subsection 3.3.4, a quick method to identify particles in dry state is proposed. This method has been adopted in order to run preliminary analysis concerning the concentration of the sample.

### 3.3.1 Identification algorithm v1.0

This algorithm has been designed in order to identify particles in videos, with the final aim to measure the direction of their motion. Conventionally, similar algorithms process videos of monodispersed samples, i.e. population of particles of the same type. In this scenario, it is possible and convenient to feed the algorithm with a priori information on the target [267], or exploit the sharpness the image to apply edge detection [268]. However, the design of this identification algorithm deals with low quality images of heterogeneous populations of the same sample, which may depict particles appearing in different configurations in the same frame. The algorithm herein proposed is tailored around the intensity values, exploiting the contrast between the dark particle and the bright background, when imaged in bright field mode. Hence, a particle can be intended as a group of low intensity pixels in a neighbourhood of high intensity pixels. In this way, the identification problem is reduced to a search for the local minima.

Before proceeding with the implementation of the algorithm, it is important to analyse the weaknesses of the input micrographs. In Section 2.6.1, both noise and uneven illumination problems have been discussed, together with some solutions. An additional problem arises from the different depth values of the particles dispersed in the media. Particles in liquid media are able to move in all directions, including perpendicularly to the detector plane. Consequently, the closer the particles to the focus plane, the darker they appear in the image. Hence, a global binary thresholding may prevent many particles from being identified in the frame.

First, the images are processed via N2V in order to remove the noise. Then, the uneven illumination is compensated by locally removing the mean value of small partially overlapping patches, as discussed in Subsection 3.2.1. In order to standardise this process, the size of the patches and the value of the stride are selected proportionally to the dimension of the input image. In the case of full size images (i.e. 3838x3710 pixels) the stride is set equal to 4 and 64 is set as side of squared patches. These values are then scaled proportionally to the size, using only integers and saturating the stride at 1 for
images smaller than 1024 pixels per side. In order to facilitate the identification process, the images are filtered with a Gaussian distribution of 3 pixels wide. This further step is adopted in order to smooth the high frequency details in the image and without losing structural information, and thus creating flat surfaces of colour.

![Diagram](image)

Figure 3.8: Schematic diagram of the identification algorithm v1.0. The images are denoised and processed to compensate the uneven illumination. The local minima are searched in order to identify the particles, and the saved coordinates are lastly refined.

At this point, the pre-processed images are divided into small areas, and the minimum value is located. This approach is applied in a raster way, covering the full image with partially overlapping patches. The so-computed intensity values are stored into a variable, together with the coordinates they occupy in the image. After processing the full image, the minima are refined in order to avoid multiple identifications, by cross-matching intensity values and coordinates. A schematic representation of the flowchart modelling this algorithm is provided in Figure 3.8. In order to reduce the number of false positives (i.e. erroneously recognised particles), different minima in a fixed neighbourhood are compared. If the difference in intensity is small or the two minima are very close, the least dark minimum is then deleted. Moreover, the path between two minima in the neighbourhood is analysed, and two different particles are labelled if a bright area is detected in between. This condition tries to model two close particles separated by the bright background (Figure 3.9).

The algorithm v1.0 identifies the particles in a frame via their darkest points. All further identifications and tracking operations will be based on the coordinates of the identified minima.

### 3.3.2 Identification algorithm v2.0

The previous algorithm presented many limitations, as emerged from the results presented in Subsection 4.4.4. However, the strongest limitation is the long execution time required to process a single full size frame (Table XIII in Chapter 4). Therefore, the
Figure 3.9: Schematic representation of the working principle of the refinement algorithm implemented to reduce false positives. On the left a full size example image, while on the right the zoomed area highlighted in the red square. An example of the refinement principle is proposed with the minima labelled as “1”, “2”, “3”. The neighbourhood centered in “1” is highlighted with a red dashed circle and includes other two minima, which are evaluated by comparison. Minimum “2” is very close to “1” and the space between them is all black, so “2” is removed. Conversely, “1” and “3” are separated by a portion of background, thus minima “1” and “3” are saved.

previous algorithm was upgraded with the aim of increasing the robustness of the identification, and reducing the processing time. This second version builds on top of the first implementation, including the same pre-processing steps. Conversely, the novelty of this upgrade relies on the local thresholding process performed by applying the adaptive algorithm proposed by Bradley in 2007 [269]. This method processes neighbourhoods as wide as one tenth of the image by subtracting their mean value. The image is then binarised, creating black blobs on a white background.

Figure 3.10: Schematic representation of the working principle of the identification algorithm implemented to increase the performances of the first version (Subsection 3.3.1). A full size example image (on the left) is locally thresholded, by dividing the image in ten sections (at the center) and then subtracting the mean values out of each of them. The resulting image (on the right) is binarised, showing black blobs corresponding to the particles in a white background.
The output is then filtered eliminating all the blobs with a radius smaller than 3 pixels. In this fashion, noise residuals or dust are prevented from being erroneously labelled, thus reducing the number of false positives (Figure 3.10). A schematic representation of the flowchart modelling this algorithm is provided in Figure 3.11. Furthermore, the processing phase becomes totally independent from the values of the pixels. Hence, the analysis results extremely fast and extracting the features of the blobs is an easy problem. Information about size, shape and location in the frame are computed and stored into a list of variables. This list of particles is no further refined, as the implementation of the algorithms generates separate blobs. Moreover, the center of mass of the blobs is computed and used to identify the location of the particles in the frame.

Figure 3.11: Schematic diagram of the identification algorithm v2.0. The images are denoised and processed to compensate the uneven illumination. Furthermore, the local minima are searched in order to identify the particles.

3.3.3 Deep LiqID

The method proposed in the previous section produces great results, it is robust and deals with local variations of illumination. However, it is possible to produce improved results by recurring to a DL-based methodology, able to fulfil complicated tasks in short processing time. The DL-based approach dominating the state of the art in the field of object detection is named YOLO, and was released by Joseph Redmon in 2016. The name stands for “You Only Look Once” and is widely used to identify objects in real time [270]. The advent of YOLO revolutionised the way of thinking the DL methods for object detection, which at that time were implemented by sliding a classifier network over patches of variable size of the images [271]. Other networks pooled similar pixels together to further pass through an image classifier [272]. This kind of approach is also known as region proposal. However, these approaches generally rely on multiple networks, which require different trainings, thus becoming slow and hard to train and optimise. Conversely, the YOLO architecture performs object detection in a single step. It is implemented as a single deep CNN that detects multiple bounding boxes coordinates and predicts multiple class probabilities (Figure 3.12).
YOLO is easy to implement, fast and produce great results. During the years its structure has been refined in order to increase the performance, leading to the release of YOLO9000 in 2017 [273] and YOLOv3 in 2018 [274]. However, after the third version of the network, Redmon abandoned the project. Only two years after, the project was continued by Bochkovskiy without any connection with the original author, and in April 2020 YOLOv4 was released [275]. Unfortunately, his glory did not last long, as in early June of the same year Jocher released YOLOv5. At the time of this thesis being written, Jocher has not released any official paper yet. Moreover, it is worth mentioning that the community receiving the latest version of YOLO debates about the legitimacy of the name “YOLOv5”, as it involves only a change of implementation, rather than adding anything new to the architecture. YOLOv5 is still under development, continuously releasing new updates and showing great results. Despite the controversies of the community between YOLOv4 and YOLOv5, in this work YOLOv5 has been adopted, as it resulted to be easy and fast to implement. The instructions provided by Jocher are available at the address https://github.com/ultralytics/yolov5.

All the different five versions of YOLO are based on the same working principle. The network predicts the four coordinates of the bounding boxes surrounding the objects as anchor boxes. The anchor boxes are bounding boxes with fixed height and width, and are used to identify scale and aspect ratio of the objects to identify. YOLO tiles the image with
thousands of anchor boxes of different shapes and size, later it identifies which anchor boxes best predict the location and the class of an object. The prediction is performed in the training phase, when YOLO computes the Intersection over Union (IoU) for every anchor box. The IoU consists in computing the portion of anchor box overlapping the ground truth, divided by the whole region covered up by the two bounding boxes (Figure 3.13) [270]. Conventionally, IoU greater than 0.5 marks a positive labelling, between 0.4 and 0.5 the prediction is ambiguous and the network does not have to identify the object, while a value lower than 0.3 stands for no object detection. The type, size and version of the anchor boxes can be defined by the user before training the network.

Figure 3.13: Schematic representation of the anchor boxes implementation. a) YOLO covers the image with thousands of anchor boxes of different sizes and shape (on the right). b) The anchor boxes are validated by computing the IoU, as the region of the predicted anchor box (in blue) overlapping the ground truth (in red) divided by the region occupied by both the bounding boxes.

In order to train the network, the user has to describe the bounding boxes surrounding the objects in the frame. Therefore, the network requires in input the x and y coordinates of the top-left corner, the width and the height of the bounding box, together with the class of the object. The first four parameters are expressed as percentage with respect to the dimensions of the image. Similarly, YOLO produces the output in the same format, two coordinates, two dimensions and the probability the identified object belongs to a specific class.

YOLO is a very flexible learning-based approach that has been applied via transfer learning to many different fields, such as tomato detection [276], face recognition [277], or identification of particles imaged via cryo-EM [243]. This last technique is named crYOLO and was trained on a wide dataset of images depicting particles embedded in ice and imaged via cryo-EM. Consequently, crYOLO identifies particles in high quality, sharp
images, which makes it perform poorly on LP EM images. Hence the need for a novel method tailored on the features of LP EM data.

In order to train the network, a database containing 780 LP EM images was defined. The database groups images of seven different specimens recorded all in TEM mode under different imaging conditions:

- 100 frames extracted from 2 different videos of cytoplasmic polyadenylation binding element (CPEB4-D4) containing about 4500 particles
- 180 frames from 2 videos of a sample containing poly(2-(methacryloyloxy) ethyl phosphorylcholine)-poly(2-(diisopropylamino) ethyl methacrylate) - also known as PMPC-PDPA - showing about 600 different objects
- 100 frames of ferritin recorded in defocus condition containing about 25000 elements
- 100 frames of ferritin recorded at low value of magnification showing about 300 particles
- 200 frames from 2 different videos of SiO$_2$ nanoparticles depicting about 2200 different objects
- 100 frames of polymersomes containing about 3500 elements
- 100 frames from 2 videos of RNA polymerase depicting about 9700 particles

The features of the database are listed in Table I for major clarity. The database was designed by adopting different specimens labelling different numbers of instances, thus creating an unbalanced database. Generally, designing a classifier requires a balanced database, in order to avoid polarising the identification towards some specific features. However, an unbalanced database in the case of one class-identifiers accounts for different intensity and shapes features of the object. One image (i.e. a single frame) from every video selected to create the database is depicted in Figure 3.14. The specimens recorded in liquid state and used to create this database will be better described in Chapter 4.

The images comprising the training have all been scaled down to 512x512 pixels in order to satisfy YOLOv5 requirements. Moreover, the noise is reduced, thus improving the quality of the features to extract. Furthermore, 20% of the images of the training set has been randomly extracted to create the test set. All the images used to train the network have been labelled by hand, by using the tool LabelImg. This software is free to use and available online at the address https://github.com/tzutalin/labelImg. It allows
Figure 3.14: The database to train the Deep LiqID is comprised of 780 different images extracted from 10 different videos. The mosaic is comprised of one frame extracted from the videos in the training set. The images have been pre-processed by applying a Gaussian filter and the contrast has been enhanced by 30% for displaying purposes.

TABLE I: Structure of the Deep LiqID training set

<table>
<thead>
<tr>
<th>Sample</th>
<th>Videos</th>
<th>Frames</th>
<th>Particles</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPEB4-D4</td>
<td>2</td>
<td>100</td>
<td>4500</td>
</tr>
<tr>
<td>PMPC-PDPA</td>
<td>2</td>
<td>180</td>
<td>600</td>
</tr>
<tr>
<td>Ferritin - defocus</td>
<td>1</td>
<td>100</td>
<td>25000</td>
</tr>
<tr>
<td>Ferritin - low mag</td>
<td>1</td>
<td>100</td>
<td>300</td>
</tr>
<tr>
<td>Nanomotors</td>
<td>2</td>
<td>100</td>
<td>2200</td>
</tr>
<tr>
<td>Polymersomes</td>
<td>1</td>
<td>100</td>
<td>3500</td>
</tr>
<tr>
<td>RNA polymerase</td>
<td>2</td>
<td>100</td>
<td>9700</td>
</tr>
</tbody>
</table>

the user to load a sequence of images and draw bounding boxes around the desired objects. Moreover, it produces in output a text file for every image, containing the coordinates of the bounding boxes and the class of every object in the format required from YOLO.

In order to improve the performances of the identification, the weights of the network have been initialised by using the pre-trained weights of the COCO dataset. The COCO dataset was created by Microsoft to train DL-based methods especially on object detection, segmentation and captioning. This dataset contains more than 1.5 millions instances, organised on 80 different classes. Thus, implementing the particle identifier on transfer learning, YOLOv5 contains already the weights necessary to correctly extract the features. However, the last classification layer of the network has been removed and replaced with a one class-output layer. Thus, training this network over the new dataset for a 1000
epochs created a novel DL-based method defined \textit{via} transfer learning to identify particles imaged in liquid state through LP EM. This implementation has been named \textit{Deep LiqID}. This DL-based approach produces in output the prediction of the location of the particles, together with the probabilities they belong to the “particle” class. The output is in the form of x and y coordinate of the top-left corner of the bounding box, the width and the height values, and lastly a value between 0 an 1 expressing for the probability.

### 3.3.4 Identification in dry state

In this subsection an algorithm to identify particles imaged in dry-state is described. This type of imaging technique produces different results than LP EM, acquiring micrographs that are easy to analyse, and characterised by better contrast than LP EM due to the use of staining agents, and the low blur as the particles are fixed and the liquid surrounding them is removed. Standard dry TEM images do not require complex processing methods, as the three approaches proposed in the previous subsections.

The method to identify particles in dry state was tailored on spherical particles of different size and ratio. Firstly, the uneven illumination was compensated as previously described in \textit{Subsection 3.3.1}. In this implementation, squared patches of 128 pixels-wide were extracted with a stride of 2 pixels in both directions. The image was also filtered by using a Gaussian filter 2 pixels wide. This filter smooths the details of the image, reducing significantly the amount of misleading information on the image, in such a way to make the identification of the particles robust. In order to simplify the particle identification step, the edges (\textit{i.e.} the sharp variations of brightness) in the image are computed, creating a mask with the profiles of the imaged elements. The edges were computed by applying the Canny method \cite{278}. At this point, the circular elements in the image were identified, saving only the elements with the two main dimensions differing at most by 20\% of their value. Consequently, the radius is therefore measured as half the average of the two main dimensions.

The identification algorithm was mainly used to measure the diameter of the particles. Unfortunately, the technique here described could not measure the membrane of imaged particles. The membrane entraps the staining agent, thus resulting dark in the images. Consequently, the application of an edge detection algorithm highlights only the transition between the membrane and the inner part of the particles. Therefore, the algorithm
was further modified in order to compensate this flaw. This implementation exploited the high density of particles in the images, and thus computing the membrane of a particle as the averaged difference of the radius of particles in the neighbourhood. Moreover, the algorithm uses only the first neighbours circling the particle of interest, excluding particles with the center further than the farthest point of the closest particle (Figure 3.15a). In addition, particles partially occupying the same trajectory are excluded. The minimum and maximum angles occupied by the neighbours are computed, and the particles overlapping this angular range of other particles are excluded (Figure 3.15b).

\[ d_m = \frac{1}{N} \sum_{i=1}^{N} \frac{d_{0i} - r_0 - r_i}{2} \]  

\( N \) is the number of first neighbours, \( d_{0i} \) measures the center-to-center distance between the reference particle 0 and the \( i \)-th neighbour, while \( r \) represents the radius of the particles. Finally, the diameter of the particle is obtained by summing the value of the size of the particle and the double of the membrane thickness, taking into account the two ends of a
diameter. A flowchart modelling the identification of particles in dry state is depicted in Figure 3.16.

Figure 3.16: Schematic diagram of identification of particles in dry state algorithm. The images are processed to compensate the uneven illumination. Furthermore, the images are blurred with a Gaussian filter in order to smooth down the details. An edge detection algorithm is then applied, before measuring the thickness of the membranes. Lastly, the algorithm writes in output the number of identified particles and a measure of their diameter.

Moreover, this identification method has been used to compute the polydispersity index (PDI), a factor measuring the heterogeneity of sizes of particles in a population [279]. The PDI is computed as the squared ratio of the standard deviation \( \text{std} \) of the diameters \( D \) over their mean value (Equation 3.6).

\[
PDI = \frac{\text{std}^2(D)}{\text{mean}^2(D)}
\] (3.6)

The image analysis software to compute the PDI of a population of particles imaged via EM was developed in Matlab and the code is available in a GitHub repository at the address https://github.com/GabrieleMarchello/PDI.

3.4 Tracking Algorithm

Particle identification algorithms were adopted in this work to localise the particles in image sequences recorded via LP EM. As previously mentioned, LP EM is a revolutionary technique due to its ability to include time in the measurements. Therefore, after identifying the particles in the micrographs an algorithm to track them in time became fundamental.

Initially, the algorithm was implemented by comparing two consecutive frames at a time. The coordinates of the particles identified in the first frame are used as center of neighbourhoods in the second frame. The algorithm searches for coordinates inside the so-defined neighbourhoods. Unfortunately, the width of the neighbourhood has to be decided \textit{a priori}, and are conventionally set equal to twice the average size of the identified particles. This approach neglects the motion of particles across the perpendicular axis, due to lack of information that can be extracted from 2D images. All the particles in the
second frame localised inside a specific neighbourhoods are then selected as candidates for the second step of the trajectory of a specific particles. Moreover, the distance between the particles in the first frame and the corresponding candidates in the second frame are computed. Consequently, the coordinates in the second frame continuing the trajectories started in the previous frame are chosen by minimising the distances. This minimisation process is based on the assumption that particles cover small portions of space over their motion across two consecutive frames (Figure 3.17). This process is further applied to all the images composing the sequence, by processing two images at a time.

Figure 3.17: Schematic representation of the tracking algorithm over time performed over two consecutive frames, separated by a time interval $\Delta t$. The coordinates of the particles identified in the first frame (at time $t$ equal to $t_0$) are used as reference in the next frame (dashed contours) and used to center the neighbourhood. A search for particles fitting the second step of the trajectories is implemented inside the neighbourhoods. The distances between the reference particle and the candidates is computed and the particles at the shortest distance is selected. Finally trajectories are created and associated to the particles in the images. The blue spots represent the coordinates of the identified particles in the first frame, while in red the coordinates in the second frame.

However, this approach presented some flaws that affect the goodness and the performance of the tracking analysis. A tracking method based on the computation of the minimum distances may not produce good results in the case of crowded environments or in the case of crossing trajectories. Moreover, since particles move independently and no global motion can be derived, every particle in the video has to be treated individually. Therefore, the search for tracking the second element of the trajectories has to explore all the possible directions.

Furthermore, in order to make the trajectory tracking robust, the minimisation distance was matched with a similarity function between the particles in the neighbourhood. Consequently, the algorithm stores the coordinates of the identified particles, together with the patches around them, thus creating a gallery of particles for every frame. Therefore, the algorithm computes the distances between one particle and the candidates in the
neighbourhood. Then, the similarity between two particles at a time is computed by applying the structural similarity index measure (SSIM). The SSIM operator analyses three different characterising factors to establish the similarity of different images: luminance \((Equation \ 3.7)\), contrast \((Equation \ 3.8)\) and structure \((Equation \ 3.9)\) [280]. In the measurement of the similarity between two different images \(I_1\) and \(I_2\), the three parameters depend on mean intensity, standard deviation and covariance \((\sigma_{12})\).

\[
l(I_1, I_2) = \frac{2\text{mean}(I_1)\text{mean}(I_2) + C_1}{\text{mean}^2(I_1) + \text{mean}^2(I_2) + C_1} \tag{3.7}
\]

\[
c(I_1, I_2) = \frac{2\text{mean}(I_1)\text{mean}(I_2) + C_2}{\text{std}^2(I_1) + \text{std}^2(I_2) + C_2} \tag{3.8}
\]

\[
s(I_1, I_2) = \frac{2\sigma_{12} + C_3}{\text{mean}(I_1)\text{mean}(I_2) + C_3} \tag{3.9}
\]

where \(C_i\) are constants included to balance the equations: \(C_1\) avoid instability, when the squared sum of the mean value of the images is close to 0, and usually \(C_3\) is chosen equal to half of \(C_2\). The luminance, contrast and structure factors are multiplied together to compute the similarity between the two pictures. The final equation is reported below

\[
SSIM(I_1, I_2) = \frac{(2\text{mean}(I_1)\text{mean}(I_2) + C_1)(2\sigma_{12} + C_2)}{(\text{mean}^2(I_1) + \text{mean}^2(I_2) + C_1)(\text{std}^2(I_1) + \text{std}^2(I_2) + C_2)} \tag{3.10}
\]

Therefore, distance and SSIM values are measured for every particle-candidate pair within the neighbourhood. The candidate with the lowest distance and the highest SSIM values is chosen as next step in the trajectory.

Once reconstructed the trajectories, the values of the positions occupied by the particles in every frame were used to compute the mean squared displacement (MSD). This measure enables to quantify the mobility of a population of nanoparticles, as it measures the variation of the registered displacement with respect to the sampling time [281]. However, given the peculiarity of the images and the samples under study, the MSD was computed by first averaging the variation of motion in every frame. These values were then used to estimate the MSD over different time intervals.
3.5 3D Reconstruction

In the previous sections, the methods designed to reduce the noise, and identify and track particles over time in LP EM images have been discussed. At this stage all the elements are in places to propose the methodology developed for performing 3D reconstruction of objects imaged via LP EM. A standardised workflow is herein proposed, describing all the steps necessary to transform a set of micrographs into a 3D model.

The proposed 3D reconstruction method sets its roots in the SPA technique, borrowing its principles and bending them to the new needs introduced by LP EM. The processing flow is implemented in Scipion, thus exploiting all the conventional reconstruction modules Scipion contains. Firstly, the image sequence has to be loaded in input, together with the parameters of the imaging conditions. It is important to correctly define the microscope voltage, spherical aberration, pixel size and magnification. The first two parameters are associated to the microscope and can be considered as constant, while both the pixel size and the magnification depends on the experimental settings. The main steps composing the workflow are defined as follows:

1. The contrast transfer function (CTF) has to be corrected in all the images, thus compensating the distortion introduced by the microscope during the imaging process. This process is performed by applying the tool CTFFIND4 developed by the Gregorieff Lab [282].

2. The recorded image sequences have to be processed in order to reduce noise and increase resolution. This step was initially left to the pre-processing tool in Scipion; however, after implementing the N2V denoising network, the pre-processing Scipion step was skipped by setting in input the images cleaned by N2V.

3. At this point that the distortion corrupting the micrographs has been suppressed, the particles are identified and localised in every frame. This task can be performed in Scipion via EMAN2, a template-based algorithm that requires user selections to generate a template of the particle to identify (Figure 3.18).

The algorithm further allows the user to refine the gallery before exporting the particles. Moreover, it is possible to filter out the background surrounding the particle, by cutting out the pixels outside a radius defined by the user. In this fashion useless or misleading information are excluded from the reconstruction. Furthermore, it is
Figure 3.18: Schematic representation of the particle selection process. The image used as example contains a population of particles of ferritin. a) The black boxes are defined by the user to generate a template of the particle. b) The particle template is used to recognise and identify all the particles appearing in the frame (green boxes). c) The particles automatically identified are stored into a gallery. Finally the template based research is applied to the rest of the images in the sequence.

4. Moreover, the extracted particles were fed in input to RELION to generate the initial 3D model, as discussed in Subsection 2.3.2. This method can be considered a reference-free technique, as it does not use any a priori information about the imaged particles.

5. The 3D model reconstructed in this fashion is further refined by applying the projection matching algorithm, in order to produce an unbiased, high-quality density map.

6. The refined 3D density map is validated by using ResMap algorithm to measure the local resolution.

The flowchart modelling the workflow to reconstruct the 3D model of particles imaged in LP EM is reported in Figure 3.19.
The present 3D reconstruction technique exploited the Brownian motion of particles captured by LP EM, using the rotational component to access all the particle profiles necessary to reconstruct the 3D model. Building on this statement, two methods have been proposed. The first method, named *Brownian Tomography* (BT), processes a single particle that is imaged long enough to show all the profiles of its structures. The second method merges BT and SPA, by exploiting the free rotation of hundreds of particles observed via LP EM. Thus, this technique is named *Brownian particle analysis* (BPA).
Figure 3.19: Schematic diagram showing all the steps involved in the workflow defined to perform 3D reconstruction. The image sequence recorded via LP EM is chosen as input. All the images are further denoised and the CTF compensated in order to increase the quality of the images. Alternatively, the sequence can be processed via N2V and used as input. The particles are then extracted by the processed images, by using or the EMAN2 module in Scipion, or the Deep LiqID presented in this work. Consequently, the particles are used to build an initial 3D model, which is further refined via projection matching. The output model is lastly processed via ResMap to measure the local resolution of the 3D model. In the flowchart, optional input are represented with a dashed line.
CHAPTER 4

RESULTS AND DISCUSSION

In the previous chapters, the innovation introduced by LP EM and the key role it can play in future scenarios of many research field have been discussed. The main difficulties associated to the imaging process and the limitations to its analyses have been clearly highlighted. In Chapter 3 the methodology developed to tackle and overcome LP EM problems has been accurately discussed. Thus, the results obtained by applying the proposed methods are presented and discussed in this chapter. The structure of the current chapter matches Chapter 3, thus starting from describing the used equipment, over to evaluating how the different employed methods perform.

4.1 Fluid Dynamic Simulation

In Section 3.1 the model of the liquid holder was reconstructed by replicating the original design of the manufacturer. The implementation in COMSOL Multiphysics replicates the actual experimental conditions and thus enables a fluid dynamics analysis of the component, in order to understand how the liquid flows and fills the liquid cell inside the holder (Figure 4.1). The velocity of the fluid in these conditions was analysed and the results are reported in Figure 4.2a. This first preliminary simulation was performed

Figure 4.1: Top view of the tip of the liquid holder simulated in COMSOL Multiphysics. The tip is flower-shaped and contains two rectangular microchips, with the observation window at the center. The tip is connected to two parallel in plane pipes, an inflow (top pipe) and an outflow pipe (bottom pipe) for flowing the liquid in the cell. An inlet port (in light blue) and an outlet port (in yellow) are set at the end of the pipes in order to simulate the application and the behaviour of an incompressible Newtonian fluid.
to validate the goodness of the model, and especially to analyse the trajectory of the flow. The first simulation confirmed that the fluid went linearly from the inlet to the outlet pipes, slowly filling the whole volume of the chamber. This phenomenon is represented by the expansion of the red streamlines in the middle of the liquid cell, towards the farther petals Figure 4.2a. Then, the complexity of the system was increased by adding the two microchips inside the flower-shaped tip, separated by a 200nm spacer. The inclusion of the microchips significantly influenced the liquid flow inside the chamber, modifying the liquid trajectory and narrowing it down to the small microfluidic channel between the microchips. Unexpectedly, most of the fluid went into the petals of the structure, generating a turbulent flow. As a consequence, the volume between the two microchips was partially filled with only a small portion of the incoming fluid. The red streamlines in Figure 4.2b highlights the lack of a clear flow trajectory, resulting affected from the turbulent flows in the adjacent petals, but not by the effect of the pump.

![Figure 4.2: a) Fluid dynamics simulations of the holder modelled without the SiN microchips. A Newtonian fluid was pumped in the inflow pipe (on the left) and out from the outflow pipe (on the right). The flow expands towards the flower shape reservoir perpendicular to the liquid main trajectory while flowing through the chamber. b) Fluid dynamics simulations of the holder comprising two microchips inside. Both simulations run applying laminar flow, planar symmetry and no slip conditions; the analysis was then performed at steady state.](image)

As a first approximation model, these results can be extended to the behaviour of the particles in solution when pumped into the inflow port, mirroring the actual experimental conditions. The simulation showed that most of the particles may not flow between the observation windows, but instead they would be trapped in the flower petal reservoir. The low intensity turbulent flow between the windows resulting from the simulation states that the motion of the particles moving over the observation window may be only Brownian, hence not affected by other undesired external components. These findings strongly sug-
gest that parameters such as sample concentration and velocity at which the sample is injected into the holder have to be carefully chosen, in order to maximize the number of the particles in the viewing window area.

4.2 Imaging Analysis

The preparation of the sample is not the only step in the whole imaging process that requires careful consideration. The entire workflow was studied in detail in order to overcome all the possible problems arising, and improve the quality and resolution of the images. The bulging effect has already been discussed in Section 3.1, highlighting the importance of focusing the beam on corners of the imaging window.

The possibility to record continuous sequence of images is pivotal to track the evolution and the motion over time of the specimen in liquid, where it is free to move. Thus, the faster the acquisition rate of the camera, the more refined the sampling of the motion. However, a high frame rate does not translate into a high quality of acquired images, as the detector has less time to be exposed to the electrons. Thus, choosing the correct frame rate to record image sequences has to be the trade off between time resolution and image quality. In the present work soft organic materials with low contrast are mainly imaged. Consequently, this type of samples is difficult to image at high frame rates. It is worth mentioning that all the image sequences analysed and processed in this thesis are the results of the PhD project of De Pace, supervised by Dr Ruiz-Perez.

In order to test the response of proteins at different frame rate values, the ferritin protein was chosen as model. Ferritin is a globular protein complex comprising 24 α-helical subunits arranged into a rigid and very stable tetracosameric structure with octahedral symmetry, forming a hollow spherical shell [283]. Ferritin is a nanoscopic cage containing multiple iron ions with an external and internal diameter of 12 and 8nm respectively, already imaged via LP EM [284]. Population of protein ferritin dispersed in phosphate buffer solution (PBS) was imaged choosing three different frame rates, 10fps, 20fps and 40fps. A single frame for every image sequence recorded at a different frame rate value has been extracted and displayed in the top row of Figure 4.3. The bottom row depicts processed images of the top row, filtered by a median filter first, and then with a Gaussian blur 3 pixels wide, in order to increase the contrast and facilitate the visualisation of the image details.
Figure 4.3: LPEM micrographs of ferritin protein dispersed in PBS, taken at different frame rate. From left to right, 10fps, 20fps and 40fps respectively. The top row depicts the raw images, while the row at the bottom an enhanced modification of the same images.

The contrast in the micrographs decreases proportionally to the frame rate, hence reducing the ability to distinguish the particles from the background, and decreasing the level of details.

Furthermore, a similar analysis was conducted by varying the frame rate and the acquisition mode of the K2 camera, in order to evaluate the best imaging conditions. Twelve different videos were recorded varying the frame rate value among 10fps, 20fps and 40fps. Half of these videos were recorded in counted mode, while the other half in linear mode. Table II contains information about the recording mode, the frame rate, the number of frames and the observation time for each video. All the videos were labelled using a progressive index. The index progression is not continuous, as the videos labelled with the missing index were discarded due to the low quality.

Quantifying the goodness of an image is a complicated task and there are no objective methods to measure it. Nevertheless, an estimation can be made by measuring the peak-SNR (PSNR), as proposed by Gonzalez [285]. This method estimates the quality of the
image by measuring the squared difference between the noisy and the original noiseless images (Equation 4.1).

\[
PNSR = \frac{\sum_{x=1}^{M} \sum_{y=1}^{N} I_N(x, y)^2}{\sum_{x=1}^{M} \sum_{y=1}^{N} [I_O(x, y) - I_N(x, y)]^2}
\] (4.1)

where \(I_O\) represents the noiseless image, \(I_N\) its noisy version, and \(x\) and \(y\) the 2D co-ordinates of the pixels in the image. Unfortunately, in LP EM the noiseless image is never available, thus the PID denoising algorithm discussed in Subsection 3.2.1 has been adopted in order to create a noiseless image to use in the method proposed by Gonzalez. The results of the application of the PID denoising algorithm are depicted in Figure 4.4.

The selection of the PID algorithm relies on its ability to produce high quality results, highlighting the signal components and removing part of the noise corrupting the images, without smoothing the details down. In order to appreciate the performance of the PID algorithm over simple denoising techniques, it is possible to compare the images on the second row of Figure 4.3, with the corresponding images in Figure 4.4. However, this algorithm is very CPU demanding and time consuming; it takes about 30 minutes to process a single frame made of 1919x1855 pixels. In Subsection 4.3.1 the performances of the PID algorithm will be discussed and analysed in details. Only a subset of three frames for every video has been selected, due to the required long processing time. These
frames were selected at the very beginning, in the middle and at the very end of each video acquired and reported in Table II. In order to prove the validity of the SNR analysis, the same measurements were repeated by using the processed images from Figure 4.3 as noisy images. The measurements of the PSNR values for each frame have then been averaged in order to obtain a more robust estimation. The results of the PSNR analysis are depicted in Figure 4.5.

The PSNR associated to every image is measured in decibel (dB) and is analysed with respect to the different imaging modes (i.e. linear or counting) and values of frame rate characterising every image sequence. However, the impossibility to accurately measure the PSNR of the image sequence by recurring to the noiseless image invalidates any quantitative analysis. Consequently, the results displayed in Figure 4.5 were used only to derive qualitative considerations. Both type of measurements clearly proved that the counted mode produces better results than the linear mode. This trend is confirmed across dif-
Figure 4.5: Graph summarising the results obtained from the analysis of image quality performed by applying the PSNR method proposed by Gonzalez. The estimation has been applied between three raw unprocessed frames extracted from each video and the PID filtered version of the same frames (in light blue). Similarly, the measurements have been repeated by replacing the raw images with their processed copies (in violet). The identification number of each video is reported on the x-axes, while the SNR values are expressed in decibel (dB) and reported on the y-axes. The SNR measurements were displayed according to the value of frame rate and the recording mode, linear or counted.

Different values of frame rate. These findings are unsurprising as the counted mode samples one pixel out of four, thus enhancing the robustness of the image to the noise.

4.3 Noise Reduction

Choosing the best image settings undoubtedly has the positive effect of improving the quality of the produced images. However, the noise corrupting the images is a problem that has to be tackled separately. In Section 3.2 several different denoising techniques
have been proposed. In this section, a discussion on the results provided by the different denoising methods is proposed.

4.3.1 End-to-end pipeline

Initially, this algorithm was first proposed to restore the uncorrupted images of poly(2-(methacryloyloxy)ethyl phosphorylcho-line)-poly(2-diisopropyl aminoethyl methacrylate) PMPC$_{25}$-PDPA$_{70}$ amphiphilic block copolymers, and poly (ethylene glycol)-b-poly(L-methionine) block copolymer PEG$_{125}$-PMET$_{120}$ vesicles, which were imaged via LP EM. PMPC-PDPA and PEG-PMET are soft materials, characterised by low density, light atoms and poor contrast, thus resulting difficult to image in their liquid native state.

The pipeline presented in Subsection 3.2.1 extends and adapts to liquid imaging by employing two of the state-of-the-art methods well established within the current image denoising and deblurring research fields. The first block of the pipeline to analyse (Figure 3.4) is the pre-processing step. In this stage of the pipeline, the images are first filtered by a median filter and then the uneven illumination is compensated, by using a patch size of 64 and a stride value equal to 2. The output of this step is further processed by a denoising algorithm. In order to find the algorithms guaranteeing the best performances, the median, the wavelet and the PID algorithms were applied to the same image of ferritin in PBS. In Figure 4.6 a comparison of these three different denoising algorithms is provided, while the processing time required to process a full size image is reported in Table III.

The PID algorithm requires about one hour and half to process a single image, a time extremely long if compared to the results of the other filters. The median filter and the wavelet denoising algorithm are two of the most common tools used in image analysis and are extremely fast, but the PID algorithm produces the highest quality images (Figure 4.6). These results can be further appreciate in Figure 4.6 from e to h, which show only a selected portion surrounding a particle in the processed imaged. It clearly emerges how the median filter produces negligible effects, and the wavelet filter completely modifies the features in the image.

<table>
<thead>
<tr>
<th></th>
<th>Median filter</th>
<th>Wavelet denoising</th>
<th>PID</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Figure 4.7</strong></td>
<td>00h 00’ 28”</td>
<td>00h 01’ 06”</td>
<td>01h 24’ 19”</td>
</tr>
<tr>
<td><strong>Figure 4.6</strong></td>
<td>00h 00’ 31”</td>
<td>00h 00’ 56”</td>
<td>01h 31’ 27”</td>
</tr>
</tbody>
</table>
Figure 4.6: Comparison of different denoising algorithms performed on ferritin imaged via LTEM in TEM mode. 

- **a)** Original raw image of ferritin as provided by the K2 camera.
- **b)** The outcome of the median filter.
- **c)** Wavelet denoising and **d)** PID algorithm. The median filter results to be not very effective, whilst the wavelet denoising removes not only the noise, but also the actual signal, resulting in a very smoothed image. Conversely, the PID algorithm removes a significant component of the noise, preserving the features in original image.

- **e)**-**h)** The area surrounding a particle corresponding to the red box in **a)**-**d)**, respectively.

Despite the PID algorithm was implemented following the suggestions of the author [260], the number of iterations has been tuned by evaluating the results obtained at different values. In Figure 4.7 an original noisy image of PMPC-PDPA is compared with the results obtained after applying the PID algorithm for 10, 20, 30 and 40 iterations, respectively. The associated processing times are reported in Table IV, averaging the results of three different executions. The level of noise decreases as the number of iterations increases. However, running the algorithm for 30 or 40 iterations did not seem to produce significant differences. Therefore, 30 iterations were selected as default value for maintaining the processing time low.

**TABLE IV: Processing time of the PID algorithm varying the number of iterations**

<table>
<thead>
<tr>
<th>Processing time</th>
<th>10 iter.</th>
<th>20 iter.</th>
<th>30 iter.</th>
<th>40 iter.</th>
</tr>
</thead>
<tbody>
<tr>
<td>00h 02’ ± 07”</td>
<td>01h 04’ ± 05’</td>
<td>01h 28’ ± 06’</td>
<td>01h 42’ ± 03’</td>
<td></td>
</tr>
</tbody>
</table>

Lastly, after mitigating the effect of the noise corrupting the images, the pipeline deblurs the denoised images, in order to restore the original sharpness. The effect of the deblurring algorithm on the image of ferritin obtained via LTEM can be further appreci-
Figure 4.7: PID results applied to a noisy image of PMPC-PDPA. (a) Original noisy image as produced by the K2 Gatan camera. (b) From left to right, the outcome of the PID algorithm at 10, 20, 30 and 40 iterations respectively. The level of noise corrupting the image reduces proportionally to the number of iterations, but above 30 iterations no significant improvements can be spotted.

Figure 4.8: Comparison on the performances of the deblurring algorithm performed on ferritin imaged via LTEM in TEM mode. (a) Original raw image of ferritin as provided by the K2 camera. (b) PID denoised image. (c) Deblurred image produced by the original method proposed by Yuan. (d) LED deblurred image.

In order to better understand and evaluate the performance of both deblurring algorithms, Yuan method and the reported LED algorithm, a region of interest (ROI) with a fixed number of pixels was chosen perpendicular to the edge of a particle of ferritin, as shown in the images displayed in Figure 4.9a. The chosen ROI was then cropped in the four images shown in Figure 4.9b. The selected ROIs were flattened along the vertical direction, hence averaging the intensity of the pixels, in order to reduce the variation of intensity. The pixel profiles were then interpolated hence obtaining the four smoothed functions shown in Figure 4.9c. The profile in blue and red, extracted from the raw and the PID denoised images respectively, result to be almost constant highlighting a mild variation in the pixel intensity close to the location of the edge. Conversely, Yuan’s method (yellow plot in Figure 4.9c) stresses the
variation of the pixels intensity between the light background with high intensity values and the dark particle of ferritin with low intensity values across the edge. However, its sinusoidal function representation results to be wide, highlighting a distortion effect due to blur, which moves the peak of the function to the further left. On the contrary, the LED technique (purple plot in Figure 4.9c) enhances the intensity values and decreases the width of the sinusoidal wave, resulting in a sharper image. In order to produce the results shown in Figure 4.9, the image was divided into areas of 128x128 pixels, with a stride equal to 8 pixels. The execution of the deblurring algorithm with these settings lasted circa three hours. The processing time was strongly influenced by the value chosen for the stride. Short processing times can be achieved by choosing small stride values, but at expense of the quality of the deblurring process. The values of the stride and the size of the area selected for the current implementation of the algorithm were set as standards for further computations.

In order to have an overview of the performance of the full pipeline, the results of the single processing steps are depicted in Figure 4.10.

Unfortunately, the effect of the bias removal step aiming at compensating the uneven illumination is difficult to appreciate, since the original image has a “flat” illumination.

In addition, the most significant outcome achieved by the proposed image analysis method arises in its ability to unveil details on images of organic materials obtained by LTEM, details that are otherwise hidden below noise and blur effects. An example of the output generated by the different stages of the pipeline is illustrated in Figure 4.11.
The noisy image in Figure 4.11a depicts PEG-PMET vesicles and micelles in solution obtained via LP EM in STEM mode. The raw images in Figure 4.11a and Figure 4.11e display aggregation of spherical nanoparticles, but it is not possible to discern whether the nano structures are membrane-bound (i.e. vesicles) or solid-core spherical structures (i.e. micelles). The application of the full pipeline including the LED deblurring algorithm highlights the presence of the vesicle membranes allowing them to be physically measurable (Figure 4.11c and Figure 4.11g). In order to ensure that the membrane features are not artefacts introduced by the deconvolution algorithm, we run two different tests. In Figure 4.11, the intermediate results of the pipeline are presented. Figure 4.11a and Figure 4.11e show the two raw images used for the membrane analysis, Figure 4.11b and Figure 4.11f the corresponding PID denoised images and Figure 4.11c and Figure 4.11g the deblurred images. Firstly, the image containing polymer vesicles and micelles in Figure 4.11a was processed, following the same procedure as described above. Figure 4.11d depicts a zoomed region of the deblurred Figure 4.11c, containing vesicles with membranes highlighted by the red halos, and micelles where there is no presence of membranes, pointed by the two red arrows. Secondly, the kernels that were estimated while processing the image in Figure 4.11c were constrained as input to the deblurring process of the image in Figure 4.11f. The result shown in Figure 4.11c clarified how a different kernel distorts the original image, without generating any membranes. Accordingly, the membranes seen in Figure 4.11g were not thought to be artefacts produced by the deconvolution, which removed only the blur introduced by the liquid nature of the sample.
Figure 4.11: Membrane analysis of PEG-PMET polymer nanoparticle obtained via LTEM in STEM mode dark field. a)-e) Original raw images as provided by the microscope detector in STEM mode. b)-f) PID filtered images obtained from a) and e), respectively, corresponding to the outcome of the first stage of the pipeline. c)-g) Deblurred images, corresponding to the output of the pipeline. d) Zoomed area of c) highlighting how the membranes is only present in the vesicular structures and not on micelles. The red halos highlight the membranes, while the red arrows point at the micelles. h) The kernel estimated during the deblurring process of c) were set in input to the deblurring process of g). The result h) is an image with a very high level of distortion and showing no presence of membranes.

4.3.2 Denoising autoencoder

The DAE proposed in Subsection 3.2.2 was designed in different stages, aiming at improving its denoising performances at every stage. The performances of all the different implementations are evaluated by processing a set of three different images fed in input. The results are presented in Figure 4.12. The selected images depict crystals formed in a sample of ferritin protein, and were recorded in full size (3838x3710 pixels).

Figure 4.12: Images of crystals formed in a sample of ferritin protein. The three images were recorded using the full size of the detector and are set in input to the DAE to test its performances.
The images result to be heavily corrupted by noise, and characterised by low contrast, which complicates the identification of the single elements in every image. The second and third images were recorded very close to the corner of the observation window, which appears in the images showing a dark sharp edge, occupying the left side of the frame. The three images were then processed by applying the denoised models obtained after training the different implementations of the network for 100 epochs, as specified in Subsection 3.2.2. The evaluation of the training loss for the three different training processes is reported in Figure 4.13 and Figure 4.14.

The comparison of the training and validation loss amongst the different architectures highlights the dependency of the number of epochs necessary to minimise the loss with the complexity of the architecture of the network. Moreover, the validation loss curves (light blue) reaches higher values than the training loss curves (orchid pink) only for low values of epochs, which implies that the denoising model can incur in overfitting if the DAE is
trained for a low number of epochs. However, the three DAE here evaluated have been trained for 100 epochs, a value high enough to guarantee low loss values - with consequent good performances - for all the three implementations. The processing times required to train the three different implementations of the DAE are reported in Table V. The results obtained by applying the different models are depicted in Figure 4.15.

![Figure 4.15: Results of the DAE denoising process applied to LP EM images in a.](image)

The original images depicted in Figure 4.12 were reported in the first row of Figure 4.15 to offer a comparison with the DAE results. The second row of results in Figure 4.15 was
TABLE V: DAE training settings

<table>
<thead>
<tr>
<th>DAE</th>
<th>N. images</th>
<th>N. epochs</th>
<th>Proc. time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Model</td>
<td>450</td>
<td>100</td>
<td>00h 18’ 32”</td>
</tr>
<tr>
<td>Batch Normalisation</td>
<td>450</td>
<td>100</td>
<td>00h 20’ 08”</td>
</tr>
<tr>
<td>Skip Connection</td>
<td>450</td>
<td>100</td>
<td>00h 41’ 29”</td>
</tr>
</tbody>
</table>

obtained with the models provided by training the basic architecture of the DAE. The images produced in output are four times smaller in size than the full size, as the basic structure was only developed to test the goodness of the denoising method. Conversely, the last two rows of results contain images in full size, in order to preserve the high resolution of the images. The third and fourth rows of images, depict the results obtained after processing the images obtained adding firstly the batch normalization and lastly the skip connections, respectively. The quality of the denoising process increases proportionally with the complexity of the architecture of the DAE. The images produced by applying the basic DAE results to be flat in intensity and shows few details. Conversely, the architecture implementing the skip connections show high contrast and high level of details, which can be particularly noticed on the two parallel dark rod-shaped crystals. The representation of these crystals appears fuzzy and shows low contrast with respect the large crystal underneath and the background. Conversely, in Figure 4.15c they are well defined and clearly emerges from the rest of the image.

4.3.3 Noise2Void

In the previous subsection, a deep learning-based approach developed to suppress noise from LP EM micrographs has been proposed. The whole architecture has been modelled and trained on a single small dataset, due to the early stage of the design of the method (Subsection 3.2.2). Conversely, the N2V is a fully developed method, proven to be efficient in different imaging conditions. In this thesis, N2V is trained in order to account for the presence of the liquid in the imaged specimen, the thickness of which varies locally in the sample holder, as discussed in Subsection 3.2.3. This factor together with the specificity of the image settings associated to each specimen severely prevent N2V from being trained once and for all on a robust and widely diversified training set. Consequently, N2V has to be trained every time that a new image sequence is produced, an approach that escalates the required processing time, and hence reduces the advantage that machine learning-
based approaches guarantee over traditional algorithms. Despite the need of a new dataset for every imaging sessions, N2V produces remarkable results on the image sequences to analyse.

The N2V has been widely adopted in this research study, and the most interesting denoising results are discussed in this subsection. However, it is worth introducing the analysis of the performances of N2V by presenting the consequences of applying a poorly structured training set. These results were produced after analysing a sequence of images of ferritin protein imaged at low electron dose, which amounts to $0.9e^{2}/Å/s$. Unfortunately, a total of only three different sequences were recorded in similar conditions. In Table VI a comparison of the metadata associated to these three sequences is provided. Thus, the denoising process could only rely on a few hundreds of micrographs, recorded in very similar conditions. The first frames extracted from each image sequence are reported in Figure 4.16.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>N. frames</th>
<th>Frame Rate</th>
<th>Mag.</th>
<th>Pixel Rate</th>
<th>Focus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence A</td>
<td>50</td>
<td>10fps</td>
<td>10kx</td>
<td>3.52Å/px</td>
<td>-11µm</td>
</tr>
<tr>
<td>Sequence B</td>
<td>100</td>
<td>10fps</td>
<td>10kx</td>
<td>3.52Å/px</td>
<td>-11µm</td>
</tr>
<tr>
<td>Sequence C</td>
<td>100</td>
<td>10fps</td>
<td>10kx</td>
<td>3.52Å/px</td>
<td>-20µm</td>
</tr>
</tbody>
</table>

Initially, only Sequence B has been included in the training process of N2V. However, the number of selected images resulted to be insufficient, thus creating an erroneous model that lowered the resolution in the original frame (Figure 4.17b). Sequence C has also been included in the training set in order to increase performances of the denoising effect, while preserving the signal in the images. Consequently, the size of the training set was doubled up, passing from an initial value of 100 images to 200. In this fashion, only the denoised versions of the 50 frames composing Sequence A have been predicted (Figure 4.17c). The results can be better appreciated in Figure 4.18, where the same patch showing two close particles has been extracted from all the three frames in Figure 4.17. Applying a model trained on a poorly designed training set may delete image features. The contours of the two attached particles at the center disappear in Figure 4.18c, thus preventing further analysis, such as particles identification, pivotal for 3D reconstructions. Conversely, the
application of a properly trained model enhances the particle features, producing a sharp representation of the two particles Figure 4.18d.

During an investigation on the motion of silica-oxide (SiO$_2$) particles dispersed in PBS buffer, dozens of different videos in similar imaging conditions were recorded. A selection of these videos was excluded from the prediction task and allocated to the training task (Figure 4.19). The imaging conditions associated to the videos selected to define the dataset are reported in Table VII. The selected image sequences can be considered as recorded at similar conditions, despite slight changes in the focus values.

Having access to a high number of image sequences recorded in similar conditions enabled the creation of a training set comprised of 700 instances. In this way, N2V can create a robust estimation of the noise distribution corrupting the micrographs. Applying the so-defined model to images recorded in similar conditions to the training set produces high
quality results (Figure 4.20). In this case, the application of N2V enhances the contrast between the particles and the background. Moreover, the noise corrupting the original image sequence smooths the edges of the nanoparticles, which result to be better defined in the N2V processed images. Improving the quality of the representation of the particles facilitate the particle identification analysis, thus producing precise measurements.

However, interesting results can still be obtained even if a proper training set cannot be designed. The solution here proposed is not recommended in case several image sequences are available and can be adopted to define the training set. This approach was used to process micrographs recorded via STEM of helix-like tubular structures generated by the block copolymer PEG125-PMET40 in PBS buffer, which was imaged at different values of magnification. All the image sequences taken into account were recorded for 10 minutes, at 10fps and are thus comprised of 7000 frames each. However, only one image sequence per
TABLE VII: Metadata associated to the image sequences of SiO$_2$ nanoparticles defining the training set

<table>
<thead>
<tr>
<th>Sequence</th>
<th>N. frames</th>
<th>Frame Rate</th>
<th>Mag.</th>
<th>Pixel Rate</th>
<th>Focus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence A</td>
<td>100</td>
<td>10fps</td>
<td>6kx</td>
<td>6.29Å/px</td>
<td>-64µm</td>
</tr>
<tr>
<td>Sequence B</td>
<td>100</td>
<td>10fps</td>
<td>6kx</td>
<td>6.29Å/px</td>
<td>-64µm</td>
</tr>
<tr>
<td>Sequence C</td>
<td>100</td>
<td>10fps</td>
<td>6kx</td>
<td>6.29Å/px</td>
<td>-62µm</td>
</tr>
<tr>
<td>Sequence D</td>
<td>100</td>
<td>10fps</td>
<td>6kx</td>
<td>6.29Å/px</td>
<td>-62µm</td>
</tr>
<tr>
<td>Sequence E</td>
<td>100</td>
<td>10fps</td>
<td>6kx</td>
<td>6.29Å/px</td>
<td>-62µm</td>
</tr>
<tr>
<td>Sequence F</td>
<td>100</td>
<td>10fps</td>
<td>6kx</td>
<td>6.29Å/px</td>
<td>-65µm</td>
</tr>
<tr>
<td>Sequence G</td>
<td>100</td>
<td>10fps</td>
<td>6kx</td>
<td>6.29Å/px</td>
<td>-65µm</td>
</tr>
</tbody>
</table>

Figure 4.19: The first frames extracted from image Sequence A (a), Sequence B (b), Sequence C (c), Sequence D (d), Sequence E (e), Sequence F (f), Sequence G (g). The frames were binned by a factor of four in order to reduce the noise and the contrast has been enhanced by 30% for displaying purposes.

value of magnification was recorded, preventing the creation of a deep and wide dataset. Consequently, the training sets were generated by sampling each image sequence every two frames, in order not to bias the training process of N2V. The result of this training process are depicted in Figure 4.21, displaying the raw images on the top row and their corresponding denoised versions at the bottom. The noise reduction effect is unsurprisingly low due to the design of the training set. Moreover, N2V sharpens the edges of the tubular structures, and enhances the contrast of the images.

Independently from the size of the training set and the goodness of the denoising process, the processing time necessary to predict the denoised version of a single image is much faster than the time required to train N2V. The results of the processing time to
Figure 4.20: **a)** The original raw frame depicting SiO$_2$ nanoparticles in PBS buffer. The frame was binned by a factor 2 in order to reduce the noise, the contrast has been enhanced by 30% and a Gaussian filter was applied for displaying purposes. **b)** The frame processed by applying the N2V trained by using the images in Figure 4.19.

train the network and to predict an image are reported in Table VIII. The table contains the data registered for the three different types of quality of the dataset discussed in this section.

TABLE VIII: Processing time for training and applying the model estimated by the N2V

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Training proc. time</th>
<th>Prediction prox. time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small training set</td>
<td>00h 41’ 22”</td>
<td>00h 00’ 03”</td>
</tr>
<tr>
<td>Wide training set</td>
<td>01h 26’ 38”</td>
<td>00h 00’ 02”</td>
</tr>
<tr>
<td>One video training set</td>
<td>02h 08’ 53”</td>
<td>00h 00’ 03”</td>
</tr>
</tbody>
</table>

The processing time associated to the training phase is proportional to the size of the image to analyse. However, the processing time associated to the prediction phase is constant and is about 2s-3s. Consequently, once the training set is built and the N2V is trained, denoising the images is an extremely fast process. This condition had the N2V become the denoising standard technique in this whole work.

4.4  **Particle Identification**

In the previous section, the results obtained by applying different methods to tackle the problem of the noise corrupting the LP EM micrographs have been provided. Mitigating
the effect of the noise had the great advantage of improving - or sometimes enabling - further analysis on the micrographs.

The methods discussed in this section were first applied to identify SiO$_2$ nanoparticles in PBS buffer, imaged in their liquid native state via LP EM. These particles were pivotal for a study concerning the motion of these particles over time, which will be discussed in Section 4.5. In this section, the results of different methods designed to identify particles are provided.

### 4.4.1 Traditional algorithm v1.0

In this section the results of the identification algorithm presented in Subsection 3.3.1 are discussed. The design of this method was based on the local minima identification, implemented via the sliding window technique. This identification technique was applied on LP EM micrographs denoised by applying the N2V approach trained on the data presented in Table VII. Additionally, the N2V filtered images are further processed by
compensating the uneven illumination by removing the bias, then applying a Gaussian filter, and lastly enhancing the contrast by 30%. The results of the different processing stages are depicted in Figure 4.22. The execution times registered while performing every preprocessing stage are reported in Table IX. The reported values are averaged on the execution time required to process all the 100 frames in a single sequence.

TABLE IX: Execution time of every stage of the preprocessing phase of the traditional algorithm v1.0

<table>
<thead>
<tr>
<th></th>
<th>N2V denoising</th>
<th>Bias removal</th>
<th>Enhancement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execution time</td>
<td>00h 00’ 03”</td>
<td>00h 06’ 29”</td>
<td>00h 00’ 48”</td>
</tr>
</tbody>
</table>

The “Original” image in Figure 4.22 is the first frame extracted from an image sequence recording the motion of SiO$_2$ nanoparticles in liquid state. This image sequence was recorded for ten consecutive seconds at 10fps, thus 100 consecutive frames are available.

The results of the identification algorithm v1.0 applied to the image sequence processed in Figure 4.22 are reported in Figure 4.23. A strong connection between the performances of the identification and the size of the selected neighbourhood clearly emerges. The performances were evaluated by varying the width of the neighbourhood among 16, 32 and 64 pixels, respectively. The number of identified objects is reported in Table X, and decreases proportionally to the size of the neighbourhood. It is important to highlight that small values of the neighbourhood guarantee high quality results in identifying small objects organised in aggregates. Unfortunately, small values of the neighbour wrongly
identify objects at the edge of the image. The results of this consideration are particularly evident in the region highlighted with a dashed red box in Figure 4.23.

However, the performance of the identification process are lowered if the image sequence to process is highly corrupted by noise and present low contrast. The image sequence analysed in Figure 4.24 is similar to the sequence shown in Figure 4.23, but the images show a general low quality. Similarly to the case of high quality images in input, the number of identified nanoparticles increases as the size of the neighbour decreases (Table X). Unfortunately, also the number of the detected false positive increases, an undesired phenomenon clearly evident in the region highlighted with a red dashed box.

Figure 4.23: The results of the identification traditional algorithm v1.0 are evaluated according to different size values of the neighbour. From left to right, the enhanced image, the results of the identification of the nanoparticles by using a neighbour wide 16 pixels, 32 pixels and 64 pixels, respectively. The coordinates of the identified nanoparticles are marked with red dots. An area with high density of nanoparticles is highlighted with a dashed red box.

Figure 4.24: The results of the identification traditional algorithm v1.0 applied on an image with low contrast and high level of noise. The identification is evaluated according to different size values of the neighbour. From left to right, the enhanced image, the results of the identification of the nanoparticles by using a neighbour wide 16 pixels, 32 pixels and 64 pixels, respectively. The coordinates of the identified nanoparticles are marked with red dots. An area with low density of nanoparticles leading to wrong identification results is highlighted with a dashed red box.

Quantifying the goodness of this identification method is an hard task, as it would require knowing the actual number of particles appearing in the images, a value unavailable a priori. Nevertheless, it is possible to manually identify 56 well defined particles in
the micrograph in Figure 4.23. A similar approach is difficult to adopt in the case of Figure 4.24, characterised by low quality. Roughly, it is possible to manually identify about 65 particles. Generally, the exact number of particles depicted in a micrograph is hard to acquire, as organic particles far from the focus plane show low contrast with respect to the background, and are difficult to distinguish when in the form of agglomerates. This scenario is worsened in the case of low quality images.

Moreover, the second set of results confirms how the edges of the images represent a problem for the identification unrelated to the quality of the images. Also, identifying objects at the very edge of an image results to be independent from the size of the neighbour. This problem is linked to the strategy at the base of the design of this identification method, which looks for local minima. Consequently, a very dark isolated pixel - or a small patch - close to an edge can easily be labelled as nanoparticle.

The processing time employed to identify the nanoparticles in a single frame varying the size of the window are reported in Table XI. The processing time values in the table were obtained by averaging the single processing times associated to the 100 frames composing the sequence.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Window 16px</th>
<th>Window 32px</th>
<th>Window 64px</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 4.23</td>
<td>00h 00’ 05”</td>
<td>00h 00’ 10”</td>
<td>00h 00’ 19”</td>
</tr>
<tr>
<td>Figure 4.24</td>
<td>00h 00’ 14”</td>
<td>00h 00’ 09”</td>
<td>00h 00’ 22”</td>
</tr>
</tbody>
</table>

As already mentioned at the beginning of the current section, this identification algorithm was firstly designed in order to track nanoparticles moving across the image sequence. Thus, the identification was applied to all the frames composing the sequence. The results of the identification process applied to three consecutive frame of the image sequence reported in Figure 4.23 and Figure 4.24 are illustrated in Figure 4.25.
The identification algorithm v1.0 is applied to three frames acquired at three consecutive time instants, labelled as $t_0$, $t_1$, and $t_2$. Top row contains the frames extracted from Figure 4.23 ($a_1$, $a_2$, and $a_3$), while bottom row the frames extracted from Figure 4.24 ($b_1$, $b_2$, $b_3$). The blue, green and yellow circles highlights the missing detection, the false motion and the false positive problems, respectively.

At a first glance, the identification algorithm seems to perform uniformly on different images in the same sequence. However, the algorithm presents some strong limitations:

- **Missing identification**: an object identified in a frame may not be identified in the next frames. This problem may be due to slight variations in the intensity of an unlabelled particle, occurring in the case of motion perpendicular to the observation plane, or towards other particles, thus creating an agglomerate. The missing identification scenario is highlighted with a green dashed circle in Figure 4.25.

- **False motion**: the location of the markers used to identify a specific object may be placed in different areas of the objects across different frames. The blue dashed circle in Figure 4.25 highlights the same nanoparticle that is localised in the frames with markers in different locations. The markers are placed at the center of the nanoparticle, at the bottom and then at the top in the three frames, respectively.

- **False positive**: the algorithm can erroneously identifies a patch of the image as nanoparticle. Furthermore, this effect is unpredictable, identifying different false
positives from frame to frame. This problem is highlighted in Figure 4.25 with a yellow dashed circle.

4.4.2 Traditional algorithm v2.0

The second version of the identification algorithm tries to overcome the limitations introduced by the original version. The useful component of the first version were saved - as the preprocessing analysis -, but the majority of the steps of the algorithm was heavily modified (Figure 3.11). Thus, the core of the identification algorithm still requires in input the enhanced version of the images to process (Figure 4.22).

This second version of the algorithm was implemented by locally thresholding the intensity of an image. Every image is first divided in ten small areas per side, which are further independently thresholded based on the mean intensity values of each area. The advantages following the adoption of a local threshold rather than a global threshold are shown in Figure 4.26.

![Figure 4.26: a) Raw image depicting SiO₂ nanoparticles. b) Result of the application of a global thresholding technique, setting the intensity value equals to 0.3. c) Results of the local thresholding technique.](image-url)

The result of the application of a global threshold technique is depicted in Figure 4.26b, and was performed by setting the threshold to 0.3, in order to obtain the best results possible. The nanoparticles occupying the central area of the image are well segmented, being isolated from the neighbours. However, the top and the left edges of the image are characterised by large areas of black pixels, which are due to the different low brightness of the top-right corner. Hence, it is impossible to distinguish the nanoparticles located in those areas. However, this phenomenon can be easily prevented by locally thresholding
the intensity values in the image (Figure 4.26c). The differences between the global and the local threshold can be appreciated in presence of uneven brightness, while the central area responds in the same way to both methods.

![Figure 4.27](image1)

Figure 4.27: Results produced by the application of the traditional identification algorithm v2.0 to images characterised by high (on the left-hand side) and low SNR (on the right-hand side).

![Figure 4.28](image2)

Figure 4.28: The identification algorithm v2.0 is applied to three consecutive frames, recorded at time $t_0$, $t_1$, and $t_2$, respectively. Top row contains consecutive frames extracted from Figure 4.23 ($a_1$, $a_2$, $a_3$), while bottom row the frames extracted from Figure 4.24 ($b_1$, $b_2$, $b_3$).

The performance of the traditional identification algorithm v2.0 can be appreciated on single frames extracted from image sequence of high and low quality, and SNR (Figure 4.27). The application of this algorithm to images characterised by high quality produces interesting results (Figure 4.28a). It does not miss a single particle characterised by high contrast, nor identifies false positives. Conversely, if the algorithm processes images with low SNR, the produced results are not very precise (Figure 4.28b). In this scenario,
the images present a low contrast and the dark gray areas are hard not to identify. Hence, the number of unidentified particles and the false positives increase. The performances of the algorithm are consistent when applied to consecutive frames, as it detects a very similar amount of objects across the whole sequence, whether applied to images with low and high SNR. The first three consecutive frames extracted from two different image sequences - characterised by high or low SNR, respectively - are proposed in Figure 4.28. The algorithm firmly identifies the major objects in every frame, despite the mild persistence of the identification of false positives. Furthermore, this algorithm identifies the objects through the coordinates of their center of mass. This localisation technique resulted to be very consistent across different frames, thus reducing the false motion problem.

4.4.3 Deep LiqID

The Deep LiqID approach was proposed in order to overcome the limitations introduced by the previously used identification algorithms, and to reduce the processing time. This approach interrupts the scheme of upgrades on the same base algorithm, by adopting a deep learning-based approach. Deep LiqID was trained for 1000 epochs on the training set (Figure 3.14), by using the COCO dataset of pre-trained weights as initial values.

The performances of Deep LiqID were evaluated in different conditions, looking for the configuration producing the best results. Initially the deep learning-based approach was trained only on 680 images, excluding the two different image sequences depicting the SiO$_2$ nanoparticles. The so defined dataset contained about 2200 instances less than its final version. The performances associated to this training process can be evaluated by analysing the graphs reported in Figure 4.29.

YOLO-based approaches are not evaluated only by measuring the error between the ground truth and corresponding predictions, but other parameters are taken into account. These indicators include Generalised Intersection over Union (GIOU), the objectness, the precision, and the recall. The GIOU is a slight variation of the IoU, which accounts for the distance between the areas to compare [286]. In the evaluation of the performance of Deep LiqID, the loss associated to the GIOU (i.e. the difference between the GIOU value and “1”) is minimised. Hence, the lower the value of the GIOU, the better the prediction. After training Deep LiqID on 680 images for 100 epochs, the GIOU for both training and validations sets drop below 0.04. The objectness, or confidence score, measures the
Figure 4.29: Deep LiqID training results, reporting the evolution over epochs of GIoU for both training set and validation set, objectness for both training set and validation set. Moreover, precision, recall and mean average precision is reported for IoU at 0.5, or spanning between from 0.5 and 0.95.

probability a box has to contain an object. The loss associated to this parameter is minimised during the training, and it reaches 0.1 and 0.15 for training and validation sets, respectively. Conversely, precision and recall are two parameters expressing the accuracy and the goodness of the identifications. The higher their values, the better the performances of Deep LiqID. Training the identification method produces a recall value very close to 1, but the precision value is about 0.8. The mAP measures the mean average precision with a specific value of IoU. Here, the mAP graphs are evaluated first by fixing the IoU value at 0.5, and then averaging the results of IoU spanning from 0.5 up to 0.95. The former reaches a value about 1, while the latter about 0.6, a value expressing how the particles are difficult to identify with high confidence score in a noisy environment.

In order to try to improve the performances of the so-trained Deep LiqID, the 100 images depicting SiO$_2$ nanoparticles were included in the training set (Figure 3.14), which then evolved from 680 into 780 different images. The weights obtained from the previous training process were set in input as starting values and the training was prolonged for
a total of 1200 epochs. The results of the so-defined training process are reported in Figure 4.30.

The first comment to propose after analysing the results depicted in Figure 4.30 concerns the peak every graph shows exactly at epoch number 1000. The peak is consistent with all the different parameters, showing an incredible deterioration of the quality of the training process. However, this phenomenon has an easy explanation: training a network with new unseen data, while using a pre-trained set of weights requires the network to be trained longer than the training process used to generate the pre-trained weights. Thus, the first 1000 epochs of the training process replicate the previous training, then at epoch 1000 a new second training phase starts. Unfortunately, the outcome of this training process it is difficult to appreciate in only 1200 epochs. Therefore, the training process has been repeated, but this time over 2000 epochs. The results of this training process are reported in Figure 4.31.
Figure 4.31: Deep LiqID training results, reporting the evolution over epochs of GIoU for both training and validation sets, and objectness for both training and validation sets. Moreover, precision, recall and mean average precision are reported for IoU at 0.5, or spanning between from 0.5 and 0.95.

The results reported in Figure 4.31 clearly show that after the peak at epoch 1000 all the parameters restart improving for the rest of the training. However, the best performances are reached at epoch 1000, while the second 1000 epochs do not enable Deep LiqID to reach high quality results. Therefore, Deep LiqID has been retrained once more over 780 images, but this time only for 1000 epochs, and using as initial values the weights associated to the COCO dataset. The results of these training conditions are reported in Figure 4.32. The processing time associated with the four different training processes is reported in Table XII, together with the training conditions.

<table>
<thead>
<tr>
<th>Training conditions</th>
<th>Processing time</th>
</tr>
</thead>
<tbody>
<tr>
<td>N. images</td>
<td>N.labels</td>
</tr>
<tr>
<td>680</td>
<td>43600</td>
</tr>
<tr>
<td>780</td>
<td>45800</td>
</tr>
<tr>
<td>780</td>
<td>45800</td>
</tr>
<tr>
<td>780</td>
<td>45800</td>
</tr>
</tbody>
</table>
The processing time strictly depends on the number of epochs and the number of images in the training set. The processing time doubles up as the number of epochs increases from 1000 to 2000. Training Deep LiqID over 780 images for 1000 epochs represents the trade-off between processing time and performance.

The model trained over 780 images for 1000 epochs is then used to predict the labels of the particles in new unseen images. The performances of the identification task depends on the confidence score set in input by the user. Conventionally, this value is chosen between 0.8 and 0.9, in order to have the object detected with very high accuracy. However, in the case of identification of nanoparticles imaged via cryo-EM, this value is lowered to 0.3 [243]. In the case of LP EM images, the noise corrupting the micrographs lowers the resolution of the objects so much that a low value of confidence score is required, as the mAP graphs in Figure 4.32 highlight. The result of the Deep LiqID predictions applied to low and high SNR images varying the confidence score from 0.1 to 0.5 are depicted in Figure 4.33.
The quality of the identification decreases proportionally to the value of the confidence score. A very low score corresponds to the highest number of identified objects, even if it implies a poor representation of the objects in the image. Of course the imaging conditions affect the results of Deep LiqID, as for a score equals to 0.1 it detects almost all the objects in the image with high SNR. Nevertheless, the results are not perfect, since few small objects are neglected. Conversely, the prediction process in images with low SNR produces consistent results, independently from the confidence score.

**4.4.4 Comparison of identification methods**

In this chapter, the results of three different methods developed to identify objects in LP EM have been presented. The methods perform differently, both in quality and processing time. Therefore, a comparison of the performances of the three methods is inevitable and useful to understand which method is the best to use.

The first parameter worth analysing is the processing time registered while identifying the objects in a micrograph made of 512x512 pixels. The choice of reducing the size of the LP EM images lowers the processing time and helps suppressing the noise. Having a fast method enables the analysis of long image sequences. The results of the registered processing times for a single frame are reported in *Table XIII*. The processing time values
The processing time values states that Deep LiqID is the fastest method, resulting about 12 and 200 times faster than the traditional algorithm v2.0 and v1.0, respectively. However, the goodness of an algorithm can not only be evaluated by its processing time, but it is necessary to understand how it performs in different conditions. In Figure 4.34, the results of the three different identification methods are shown, in order to evaluate and compare their performances in case of images with high and low SNR.

Figure 4.34: Comparison of the performances produced by analysing images with low and high SNR. The results are produced over an enhanced version of N2V denoised images, by applying the identification algorithm v1.0, v2.0 and Deep LiqID.

The three proposed methods generally perform well in the case of high SNR images. However, it is possible to notice that the traditional algorithm v1.0 misses a few particles, and identifies as particles very small black dots at the edges of the image, thus increasing the number of false positive. Conversely, the identification v2.0 method does not miss a single particle, outperforming even Deep LiqID, which misses some low detailed particles, or agglomerates of particles, as in the top-left side of the image. Evaluating the perfor-
mance of the three identification methods in the case of low SNR images is a hard task, as distinguishing particles from noisy patches is difficult also for human operators. Thus, rating the performance of algorithms without ground truth may result not very accurate. However, it is possible to appreciate that Deep LiqID produces the best results in the case of identifying particles in images with low SNR. It still misses a few particles, but skips patches of the image with a very low contrast and level of details, which could be just noise or particles very far from the detector. The identification algorithm v1.0 identifies only particles with high contrast, but it presents its common limitations, as high value of false positive and detection of particles along the edges. On the contrary, the identification method v2.0 recognises practically every dark patch in the images, unfortunately identifying even patches of pure noise, as the three red dots in line at the center of the image. In conclusion, v2.0 reaches the highest performance in the case of high quality images, but performs poorly in the case of low quality ones. Conversely, Deep LiqID produces steady, great performances, proving to be robust and independent to the level of noise corrupting the images. Moreover, it is important to mention that the performances of both traditional algorithms cannot be modified, while Deep LiqID can improve and produce progressively better results. In order to do so, the dataset used to train Deep LiqID can be enriched, by adding hundreds of labelled images, and/or creating different classes to label the dataset.

4.4.5 Identification in dry state

Identifying particles in dry state is a much easier task than the identification of particles in liquid state, as the absence of fluid surrounding the particles both increases the contrast and removes the motion blur component at the same time. Thus, the Canny edge detection is adopted in order to identify circular objects. The current identification algorithm is tailored to very specific imaging conditions that plays a small role in this thesis. However, this identification tool has been applied and validated in an external research project. In this work, Apriceno et al. study the phenomenon of selective targeting by analysing the behaviour of poly(ethylene glycol)-poly(2-(diisopropyl amino) ethyl methacrylate) (PEG-PDPA) in the formation of bonds. PEG-PDPA creates spherical polymersomes that have to be stained in order to be imaged in TEM, due to low density [279]. The images in Figure 4.35a1-Figure 4.35b1 show structures formed by PEG-PDPA of different sizes stained with PTA for two and five seconds, respectively. The higher
the time, the darker the colour of the staining agent in the images. Thus, the stained particles show to the detector a difference in the intensity of the contrast with respect to the background. Low contrast makes the identification hard, blurring the edge between particles and background. In **Figure 4.35a2-Figure 4.35b2** the results of the identification process are displayed.

Figure 4.35: **a1)**-**b1)** Raw images of PEG-PDPA imaged in TEM and stained with PTA. **a2)**-**b2)** Results of the identification algorithm with the identified elements surrounded with red bounding boxes. **a3)**-**b3)** The histograms derived from the distribution of the diameters computed from the identified particles.

A graphical interpretation of the results of the identification algorithm is depicted in **Figure 4.35a2-Figure 4.35b2**. The identified particles are surrounded by red bounding boxes. The number of particles identified in the the top and bottom images vary significantly, counting 537 and 175 particles, respectively. However, the algorithm seems to behave uniformly, despite the variation in number and size of the particles contained in every image. In particular, the algorithm identifies a very high number of particles, even if it suffers in the case of particles with low contrast. Therefore, the performances of this algorithm are characterised by the presence of both false positive and true negative (i.e. unlabelled particles). These side effects are independent on the size of the particles, thus representing a systematic error. However, it is worth mentioning that the algorithm ignores the deposit of staining agents - the very dark patches in **Figure 4.35a2**-, but processes
all the bright round elements in the images, such as the values of the scalebars (which, of course, could not be avoided). Lastly, the size of the identified elements is measured in order to compute the PDI, as modelled in Equation 3.6. A graphical representation of these data is produced in the form of histograms, that highlight the distribution of the diameter values of the particles (Figure 4.35a-Figure 4.35b). The two graphs differ in the number of particles included, but show similar trends that thus result in similar values of PDI: 0.26 and 0.27 for the top and bottom processed images, respectively. These results match the performance obtained by applying standard techniques, as DLS [279].

4.5 Particles motion tracking

The methods designed to identify particles imaged in liquid state were a fundamental piece in the process to observe their motion. This analysis was firstly applied to hollow nanoparticles of SiO$_2$ asymmetrically coated with urease [287]. The biocatalytic reaction between urease and urea generates a gradient of chemical species (CO$_2$ and NH$_3$), thus propelling the nanoparticle. In this fashion, the urea may act as a fuel for the silica nanoparticles, significantly affecting the motion. Therefore, applying the identification algorithms to all the frame composing the image sequences enables the reconstruction of the trajectories of the nanoparticles over time. In order to evaluate the consequences generated by the adoption of a particular identification method, the three algorithms were applied to two different sets of image sequences depicting SiO$_2$ nanoparticles coated with urease diluted with PBS at a concentration of 1mg/mL. The two sets are characterised by low (Figure 4.36) and high (Figure 4.37) SNR, respectively. This two-levels analysis matches the structures designed to evaluate the single methods.

The initial and the final time points of each image sequence are reported in Figure 4.36a and Figure 4.36b for the case of low SNR, and Figure 4.37a and Figure 4.37b for the case of high SNR. In order to have a quick estimation on the motion of the imaged particles, all the 100 frames composing the sequence have been averaged and the results are depicted in Figure 4.36c and Figure 4.36c. The particles appearing dark and well defined in these averaged images are the ones that moves the least. On the contrary, the particles that move fast and cover much space almost disappear, as the position changes in every frame. These considerations can be further and better appreciated after reconstructing the particle trajectories over time. The results obtained by applying the identification v1.0, v2.0
Figure 4.36: a)-b) Initial ($t_0$) and final ($t_f$) frame of the image sequence depicting SiO$_2$ nanoparticles coated with urease, and diluted with PBS at 1mg/mL. The image sequence is characterised by low SNR. c) Projection obtained by averaging all the images composing the sequence. d)-e)-f) Trajectories reconstructed by connecting the positions of the identified objects by applying the traditional algorithm v1.0, v2.0, and Deep LiqID, respectively. Every trajectory is represented with a specific color.

and Deep LiqID to low and high SNR images are depicted in Figure 4.36d-Figure 4.37d, Figure 4.36e-Figure 4.37e, and Figure 4.36f-Figure 4.37f, respectively.

The trajectories represented in Figure 4.36 and Figure 4.37 have been filtered, displaying only particles identified for at least 30 different frames. This condition tries to limit the number of false positives and tracking dust and noise components. The trajectories of all the particles are represented with a different colour. Colour coding the trajectories has the double effect of identifying the motion of a single identified object and, at the same time, highlighting the limits of the identification methods. In particular, the traditional algorithm v1.0 reconstructs a very high number of objects across the image sequences, still presenting several objects close to the edges of the frames, which in Subsection 4.4.1 has been presented as a problem of false identification. Identifying a high number of objects of course does not correlate to a false identification problem. However, the traditional algorithm v1.0 labels a high number of objects in the left area of the image sequence
Figure 4.37: a)-b) Initial ($t_0$) and final ($t_f$) frame of the image sequence depicting SiO$_2$ nanoparticles coated with urease, and diluted with PBS at 1mg/mL. The image sequence is characterised by high SNR. c) Projection obtained by averaging all the images composing the sequence. d)-f) Trajectories reconstructed by connecting the positions of the identified objects by applying the traditional algorithm v1.0, the traditional algorithm v2.0, and Deep LiqID, respectively. Every trajectory is represented with a specific color.

In this case, the results provided by the algorithm v1.0 are scarcely trustworthy (Figure 4.36d). In fact, the algorithm v2.0 identifies a lower number of objects in the very same area (Figure 4.36e). These results are vigorously stressed by the application of Deep LiqID that ignores that area(Figure 4.36f). Moreover, the algorithm v2.0 shows a significant improvement, by identifying also a low number of particles close to the edges. One problem all the three different methods show in the case of image sequences with low SNR is related to the overlapping trajectories (Figure 4.36). If one or more labelled objects are close and difficult to clearly identify, it results challenging for the proposed methods to keep the trajectories separated over time. However, this problem almost disappear as the quality of the image sequences increases (Figure 4.37).

A further consideration worth mentioning concerns the problem presented as “false motion” in Subsection 4.4.1. This problems is particularly effective in the case of the
analysis of the reconstructed trajectories. The identification algorithms are expected to assign to a still particle the very same coordinates in every frame. Otherwise, a motion component will be used to describe the behaviour of a particle. Hence, the less the particles move, the smaller the area occupied by the colour-coded trajectory. Unsurprisingly, the traditional algorithm v1.0 produces the lowest performances, associating fuzzy trajectories to still particles. Moreover, this problem can be analytically analysed by computing the MSD associated to every imaged sample. The MSD values measured for the trajectories depicted in Figure 4.36 and Figure 4.37 are depicted in Figure 4.39.

The MSD values were plotted by processing the coordinates registered by the traditional algorithm v1.0, v2.0 and Deep LiqID, displayed in light blue, red and yellow, respectively. The MSD data need to be interpreted as they show similar results, despite processing very different scenarios. The image sequence characterised by low SNR recorded still particles, as the trajectories showed in Figure 4.36. Thus, the MSD curves were expected to results almost flat, hence accounting only for small oscillations around the identification points. Moreover, the very first MSD value measured by all the three methods result about 1000px², a value modelling the aforementioned false motion problem. This problem seems to be mitigated in the case of high SNR image sequence, as the identification task is easier to perform. However, in this case the raising trend is not an
undesired effect, but takes into account the motion shown by the particles imaged in the top area of Figure 4.37.

The problem of the false motion results to be mitigated in the case of high quality images, and significantly reduced when traditional algorithm v2.0 or Deep LiqID are applied (Figure 4.39). However, it appears to significantly affect the study of the motion of all the imaged particles. Hence, it is very hard to understand if the imaged populations is composed by particles that do not move. In order to compensate these problems and produce a reliable method to track the motion of particles in liquid, other techniques have to be employed. Tailoring the design of the algorithm on the template of the object to track would increase the confidence of the identification, but significantly lower the flexibility of the software. Thus, a possible solution could be implemented by recurring to learning-based methods. In particular, it would be possible to extend Deep LiqID, exploiting the identification skills to track objects over time. A similar approach has been adopted by Dong et al., merging YOLO and long short-term memory (LSTM) NN to detect and track multiple objects [288]. Unfortunately, this approach was drafted at the very end of this research project, with an insufficient amount of time left to implement it. Thus the tracking analysis has been developed, but unfortunately not further applied.

4.6 3D reconstruction

In the first chapters of this thesis, 3D reconstruction has been presented as the main line of work undertaken during this research work. In this section, the outcome of all the methods previously discussed and main achievements in the field of 3D reconstruction are presented.
Figure 4.40: a) The first frame extracted from the image sequence depicting the disk-like micelle structure formed by PMPC-PDPA copolymer in solution. b) Cartoon modelling the molecular structure of the PMPC-PDPA chain (top) and disk (bottom). c) Disk-like micelle profiles extracted from the image sequence recorded via LP TEM, uniformly sampled every 8.2 seconds, in order to produce a detailed gallery of the particle motion. All the different imaged images show different orientations and occupying different positions.

Initially, the reconstruction flow (Figure 3.19) was tested on the image sequence shown in Figure 4.40a, which depicts a few structures generated by the assembly of the copolymer PMPC-PDPA. This copolymer can assemble into disk-like micelles with an average thickness of about 15nm and radius from up to 30nm (Figure 4.40b). Figure 4.40a is extracted from the image sequence recorder via LP EM and depicts only a few objects. Two of them occupy the top-right and the bottom-left corners of the image. A third disk clearly appears at the center of the image, and it was used to perform the BT reconstruc-
tion technique. The image sequence was recorded via LP EM for almost 156 seconds at 10fps, thus counting about 1500 frames.

The imaged disk is free to move and rotate in the liquid media, thus showing to the camera always a different profile. In Figure 4.40c, a selection of these different orientations captured by the camera is provided. The selected profiles have been extracted by uniformly sampling the full image sequence every 10 seconds, in order to produce a precise overview of the motion. These imaging settings enable the reconstruction of the 3D volume of the single imaged structure, by applying the methodology discussed in Section 3.5. This analysis produced the initial coarse model (Figure 4.41a) and its refined version (Figure 4.41b), characterised by a resolution value of 1nm.

Each profile is completely random and the result of Brownian rotation, as showed by the time-coloured palette in Figure 4.42a, that reveals the stochastic evolution of each profile as a function of the observation time. Specifically, motion refers to Brownian motion in confinement, due to the physical characteristics of the liquid cell. However, the present study does not distinguish between the motion experienced by the particles in the case of being located close to the observation windows or in the bulk. The motion of the particles is exploited for the sole purpose of visualising the particles views. During the refining phase, an angular orientation in 3D is assigned to each 2D profile. In Figure 4.42b the angular assignments are plotted around the refined 3D model. The assigned orientations have been used to compute the mean angular squared displacement (MSAD), an operator measuring

Figure 4.41: Results of the 3D reconstruction process of the disk-like micelle. a) Initial coarse model. b) Refined model.
the evolution of angular displacement over time. The result of this analysis shows a flat curve, implying that the acquisition rate is too slow to track the rotation of the disk (Figure 4.42c). Additionally, the MSAD was used to measure the rotational coefficient, which was further compared to the theoretical value (Table XIV). Unfortunately, the measured rotational coefficient results to be three orders of magnitude lower than the theoretical value. Temporal resolution of 100ms is four orders of magnitude longer than the rotational relaxation times of the structure, which is in the range of microseconds. The difference between the afforded temporal resolution and rotational relaxation times hinders an effective tracking of the angular dynamics. Yet, it is possible to capture a large number of profiles and use them sequentially to reconstruct the full 3D structure.

**TABLE XIV: Disk-like micelle rotational coefficient**

<table>
<thead>
<tr>
<th>Rotational coefficient</th>
<th>Theoretical [rad$^2$/s]</th>
<th>Measured [rad$^2$/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rotational coefficient</td>
<td>$10^3 \div 10^4$</td>
<td>$\sim 10^1$</td>
</tr>
</tbody>
</table>

Figure 4.42: a) Time-coloured palette sequences of a PMPC-PDPA disk-like micelle in solution imaged sequentially at 10fps, with each colour representing a time point. b) MSAD computed by tracking the angular orientations assigned during the 3D reconstruction process. c) Azonometric projection of a PMPC-PDPA disk-like micelle shown alongside the different orientation collected as a function of time.
The reconstruction process produced interesting results, reaching resolution of *circa* 1nm, a value almost low enough to visualise the building block, with the PMPC and PDPA measuring 2.8nm and 1.7nm, respectively [289]. Therefore, the reconstructed volume has been further process by Dr Acosta Gutierrez, an external collaborator, in order to derive structural information by combining the experimental data with modelling. A coarse model of a single PMPC-PDPA chain has been modelled by using different units – or beads – that comprise the critical chemical elements, allowing us to reduce each polymer from 2536 atoms to 316 beads. The use of the coarse model allows us to simulate the full disk (1 million atoms) and capture the full dynamics of the system including its self-assembly behavior. The disk-like micelle structure has been simulated, using the number of PMPC-PDPA chains as free parameter to identify [290]. A comparison of the simulated model and the BT reconstruction is shown in *Figure 4.43a*. The PMPC-PDPA chains in the middle of the structure results to be slightly stretched, while the chains on the disk edge are collapsed and coiled up (*Figure 4.43b*).

*Figure 4.43: a) Comparison between the simulated model and the BT reconstruction. b) End-to-end distance of PMPC-PDPA chains from the core and edge areas of the disk-like micelle. The chains highlighted in the black boxes have the PMPC part coloured in pink, while the PDPA component in blue.*

The BT reconstruction technique proved to be a reliable tool to investigate the structure of a single object imaged via LP EM. However, the performance is deeply dependent on the number of profiles that can be extracted out of the image sequence. Unfortunately, in the case of soft materials long exposure time is unsustainable, because of the beam damage effect. BPA can represent a solution to this problem, as it extracts profiles from different particles recorded in the same image sequence and thus requiring a lower obser-
vation time than BT. In the case of PMPC-PDPA, BPA cannot be performed, since only few objects are recorded in the image sequence (Figure 4.41).

The very first BPA analysis was performed on a specimen of protein ferritin (Figure 4.44). Specifically, a sample of equine spleen apoferritin was diluted with PBS at a concentration of 2mg/ml. The sample was imaged via LP EM for five consecutive seconds at 10fps, thus generating 50 frames at maximum resolution (i.e., 3838x3710 pixels). The image sequence was recorded at high electron dose of $13.9 \times 10^{-4}/\text{A}^2$ per frame, resulting in a total dose of circa $700 \times 10^{-4}/\text{A}^2$. The images were recorded slightly overfocus, in order to increase the contrast in the images. In this fashion, the nanoparticles appear as dark circles with a very bright center. The imaging conditions are summarised in Table XV.

**TABLE XV: Metadata of the image sequence of ferritin protein used in the BPA analysis**

<table>
<thead>
<tr>
<th>Sequence</th>
<th>N. frames</th>
<th>Frame Rate</th>
<th>Mag.</th>
<th>Pixel Rate</th>
<th>Focus</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Figure 4.44</em></td>
<td>50</td>
<td>10fps</td>
<td>10kx</td>
<td>1.12 A/px</td>
<td>20 µm</td>
</tr>
</tbody>
</table>

These images were imported in Scipion and processed by recurring to the built-in processing tool, as at the time this first reconstruction analysis was performed N2V was not available yet. The particle picking task was performed by applying EMAN2, a supervised training algorithm requiring the users to manually select few particles to create a template of the ferritin nanoparticles in order to autonomously identify the other elements. The data associated to the particle picking process are reported in Table XVI.

**TABLE XVI: Parameters used for the particle selection step**

<table>
<thead>
<tr>
<th>EMAN2 particles picking</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Training frame</em></td>
</tr>
<tr>
<td>Manually picked particles</td>
</tr>
<tr>
<td>Extracted profiles</td>
</tr>
<tr>
<td>Window size [px]</td>
</tr>
<tr>
<td>Particle size [px]</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>15</td>
</tr>
<tr>
<td>5241</td>
</tr>
<tr>
<td>150</td>
</tr>
<tr>
<td>100</td>
</tr>
</tbody>
</table>

The image sequence shows a high number of particles, most of which are well defined and isolated. Aggregation of particles can be spotted in all the corners of the frame, but the bottom right corner. EMAN2 identified about five thousands particles profiles across the whole image sequence, which were obtained by imaging only 120 different particles in
the field of view. The selected profiles were then passed in input to the reconstruction steps, in order to obtain an initial model first (Figure 4.45a), and its refined version later (Figure 4.45b). Both models were reconstructed by following the procedure defined in Section 3.5 and the associated processing time are reported in Table XVII. The goodness of this reconstruction can be appreciated by comparing the BPA model generated by LTEM data to that obtained via x-ray tomography (Figure 4.46).

Figure 4.45: a) Initial coarse model of ferritin protein obtained via BPA. b) Refined 3D model of ferritin protein obtained via BPA.
TABLE XVII: Processing time associated to the 3D reconstruction of ferritin protein

<table>
<thead>
<tr>
<th>Processing time</th>
<th>Init. model</th>
<th>Ref. model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>00h 05’ 02”</td>
<td>02h 38’ 03”</td>
</tr>
</tbody>
</table>

Figure 4.46: a) Comparison of the cross section of the 3D model of ferritin obtained via BPA, and x-ray tomography, represented as density map or cartoon. b) Side to side comparison of half of the 3D reconstruction obtained via BPA (on the left) and x-ray tomography (on the right), represented with the density map covering the cartoon skeleton.

It is interesting to notice how the BPA model matches in size and shape the model obtained from x-ray data. Both density maps appear as hollow spheres, circa 12 nanometers wide and with the shell measuring 2 nanometers thick. The 3D map obtained as result of BPA is associated to a mean and median resolutions of 4.74Å and 4.80Å, respectively. The resolution values were measured by applying ResMap. The computed resolution values are close to the those obtained via cryo-EM, which registered the lowest resolution of 1.25Å [291]. Consequently, these first results represented a starting point for reconstructing density maps of particles imaged in their liquid native state with a very low resolution. Hence, the investigation was pushed into the time domain. The five second long image sequence was split into five consecutive not-overlapping sub image sequences, each covering one second of observation. The data associated to the different intervals are reported in Table XVIII. This approach provided an estimation of the effect time plays during the
observation process, how and if the particles in the specimen evolve. More importantly, this approach allows the possibility to appreciate these events in the structure of the imaged structures. The reconstructions obtained are reported in Figure 4.47. The reconstructed 3D maps are colour-coded in order to stress the time variation.

TABLE XVIII: Data associated to the 5 sub-sequences for time investigation in ferritin protein structure

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Time interval</th>
<th>N. frames</th>
<th>N. profiles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seq01</td>
<td>0s-1s</td>
<td>10</td>
<td>1076</td>
</tr>
<tr>
<td>Seq12</td>
<td>1s-2s</td>
<td>10</td>
<td>1033</td>
</tr>
<tr>
<td>Seq23</td>
<td>2s-3s</td>
<td>10</td>
<td>1205</td>
</tr>
<tr>
<td>Seq34</td>
<td>3s-4s</td>
<td>10</td>
<td>950</td>
</tr>
<tr>
<td>Seq45</td>
<td>4s-5s</td>
<td>10</td>
<td>977</td>
</tr>
</tbody>
</table>

Figure 4.47: Initial (top row) and refined (bottom row) models of ferritin protein obtained by processing the profiles extracted from five different not-consecutive time intervals.

Unsurprisingly, the initial models seem all pretty similar, highlighting a spherical structure for the ferritin protein. Moreover, this generic spherical structure is matched by the corresponding refined models. The resolution of the five refined models was measured via ResMap, and the computed distribution are reported in Figure 4.48. The distributions result to be very similar, only the distribution associated to the first model (dark red) slightly differs, as it shows a low and wide area close to the peak. The mean and median
resolution values are reported in Table XIX. The resolution values remain almost constant across time, also resulting independent to the number of involved profiles.

**Table XIX: Resolution values associated to the ferritin protein reconstructed models**

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Time</th>
<th>N. frames</th>
<th>N. profiles</th>
<th>Mean res.</th>
<th>Median res.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seq01</td>
<td>0s-1s</td>
<td>10</td>
<td>1076</td>
<td>5.42 Å</td>
<td>5.50 Å</td>
</tr>
<tr>
<td>Seq12</td>
<td>1s-2s</td>
<td>10</td>
<td>1033</td>
<td>5.39 Å</td>
<td>5.40 Å</td>
</tr>
<tr>
<td>Seq23</td>
<td>2s-3s</td>
<td>10</td>
<td>1205</td>
<td>5.39 Å</td>
<td>5.40 Å</td>
</tr>
<tr>
<td>Seq34</td>
<td>3s-4s</td>
<td>10</td>
<td>950</td>
<td>5.37 Å</td>
<td>5.40 Å</td>
</tr>
<tr>
<td>Seq45</td>
<td>4s-5s</td>
<td>10</td>
<td>977</td>
<td>5.37 Å</td>
<td>5.40 Å</td>
</tr>
<tr>
<td>Seq05</td>
<td>0s-5s</td>
<td>50</td>
<td>5241</td>
<td>4.74 Å</td>
<td>4.80 Å</td>
</tr>
</tbody>
</table>

Despite the fact that both the mean and median resolution values do not vary in time, the 3D maps show a number of missing patches increasing in time. It may appear that these patches depend on the number of profiles, a theory nullified by the third 3D model, comprised of the highest number of profiles (i.e. about 1200), but showing a high number of missing patches.

The first attempt to investigate and discover the reasons behind the patches missing from the structures aimed at the electron dose used to record the image sequence. Conventionally, a high electron dose value is at the base of the undesired effect of the beam damage, which can be minimised by lowering the electron dose value. Unfortunately, the threshold between high and low dose depends on many parameters, such as the nature of the specimen, the temperature, and the media of the sample. In the case of protein imaged via cryo-em, the threshold is set to a few electrons per angstrom squared per second [292]. However, this threshold can be raised when the sample is imaged in solution, hence via LP EM, due to the high conductivity of the liquid nature of the

![Figure 4.48](image_url)  
*Figure 4.48: Schematic representation of the resolution distribution associated to the ferritin refined 3D models depicted in Figure 4.47. The resolution distributions are colour-coded as the 3D maps in Figure 4.47.*
PBS that lowers the electric charge of the particles [293]. Thus, a new image sequence recording ferritin protein was recorded at a dose almost 70 times smaller (i.e., about $0.2e^-/\AA$ per frame), while maintaining the other experimental parameters unchanged, such as sample concentration, magnification, and acquisition rate. Moreover, the image sequence was recorded for only five seconds, in order to reduce the cumulative electron the particles in the specimen receive. However, low electron dose directly translates into low contrast, resolution and a general low quality of the images, as discussed in Section 4.2. The first frame of the image sequence recorded at low dose is depicted in Figure 4.49 and the imaging conditions are reported in Table XX. Identifying particles in such image sequences is a very hard task, thus defining a significant role for the pre-processing step.

\textit{TABLE XX: Metadata associated to image sequence of ferritin protein recorded at low dose}

\begin{tabular}{|c|c|c|c|c|}
\hline
Sequence & N. frames & Frame Rate & Mag. & Pixel Rate & Focus \\
\hline
Low Dose & 100 & 10fps & 10kx & 1.20 $A/\text{px}$ & -15 $\mu$m \\
\hline
\end{tabular}

Initially, before developing N2V or similar denoising techniques, the frames composing the image sequence were averaged at different sampling values. Since the image sequence is composed of 50 frames, the averaged frames were obtained by averaging 2, 5 and 10 consecutive frames. The results of this process are reported in Figure 4.50. Averaging consecutive frames is a technique that in this research work has always been discouraged, as it vanishes the contributions given by the observation of the motion. However, in this specific case, the averaging technique can be used in order to obtain an estimation of the goodness of the reconstruction strategy, as the particles resulted to be very still.

Unsurprisingly, the contrast and the ability to appreciate the particles increase proportionally to the number of frames averaged together, from Figure 4.50a to Figure 4.50c.

\textit{Figure 4.49: First frame extracted out of the image sequence depicting ferritin protein recorded at low dose (about $0.2e^-/\AA$ per frame). The frame was filtered using a Gaussian filter and the contrast was enhanced for display purposes.}
The obtained image sequences were processed to reconstruct 3D models of ferritin protein imaged in low dose. The associated data are reported in Table XXI, with the electron dose value referred to each frame. It is worth noticing that despite the sequence obtained by averaging 2 consecutive frames at a time contains half of the frames of the original image sequence, it contains 75% of the amount of profiles extracted from the original image sequence. This high number of profiles available is due to the high contrast and low noise resulting from the averaging process. The 3D models obtained from these image sequences are reported in Figure 4.51.

The model reconstructed with the image sequence recorded at high dose in Figure 4.44 is shown in Figure 4.51 as reference. The 3D model reconstructed by analysing the original sequence imaged at low dose roughly maintained the spherical shape, but it presents a very high number of missing patches, resulting mainly comprised of dust. The reconstruction improves as the number of averaged frames increases. Similarly, the resolution measured by applying ResMap improves, a surprising result since the number of selected profiles significantly decreases (Table XXI).
Figure 4.51: a) Ferritin reconstructed model imaged at high dose. b) Model obtained from the original image sequence imaged at low dose, further modified by averaging 2 (c), 5 (d) and 10 (e) consecutive frames at a time, respectively.

TABLE XXI: Data associated to image sequences of ferritin protein recorded at low dose and averaged to produce 3D maps

<table>
<thead>
<tr>
<th>Sequence</th>
<th>$e^-$ dose</th>
<th>N. frames</th>
<th>N. profiles</th>
<th>Mean res.</th>
<th>Median res.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original seq.</td>
<td>0.2$e^-$/Å</td>
<td>50</td>
<td>5950</td>
<td>5.59 Å</td>
<td>5.60 Å</td>
</tr>
<tr>
<td>Avg. x2</td>
<td>0.4$e^-$/Å</td>
<td>25</td>
<td>4528</td>
<td>4.96 Å</td>
<td>5.10 Å</td>
</tr>
<tr>
<td>Avg. x5</td>
<td>1$e^-$/Å</td>
<td>10</td>
<td>1718</td>
<td>4.77 Å</td>
<td>4.80 Å</td>
</tr>
<tr>
<td>Avg. x10</td>
<td>2$e^-$/Å</td>
<td>5</td>
<td>1046</td>
<td>4.76 Å</td>
<td>4.80 Å</td>
</tr>
</tbody>
</table>

The 3D model reconstructed from cryo-EM has been superimposed to the x-ray structure, in order to produce a reference to BPA reconstructed 3D maps (Figure 4.53). This comparison shows the high quality and precision cryo-EM reconstructions can reach, with the helix structures clearly distinguishable. Additionally, the reconstructed 3D models were further investigated in order to try to understand the goodness of BPA, and especially the goodness of BPA applied to image sequences recorded at low dose. The ferritin protein 3D maps reconstructed via BPA were compared with the x-ray model, computing the correlation between both models, in order to validate and estimate the goodness of
Figure 4.52: **a-f)** Superimposition of the x-ray and the BPA models reconstructed from the image sequence recorded at high dose, including the 5 models extracted from one-second long intervals and the 3D map obtained from the full sequence. **g-i)** Superimposition of the x-ray and the 3D maps reconstructed from the image sequence recorded at low dose binned 2, 5 and 10 times.

The results of the measured correlation are reported in Table XXII, together with the number of profiles involved in the 3D reconstruction process and the associated mean resolution. The 3D maps reconstructed from image sequences recorded at high dose show a correlation generally about 0.7, with a peak of 0.75 derived from the second time interval. Similarly, the 3D maps corresponding to image sequences recorded at low dose reports similar correlation values, with the only exception of the sequence resulting from the
### TABLE XXII: Correlation between the BPA models and the x-ray models

<table>
<thead>
<tr>
<th>Sequence</th>
<th>N. profiles</th>
<th>Mean res.</th>
<th>Correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>High dose</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq01</td>
<td>1076</td>
<td>5.42Å</td>
<td>0.64</td>
</tr>
<tr>
<td>Seq12</td>
<td>1033</td>
<td>5.39Å</td>
<td>0.75</td>
</tr>
<tr>
<td>Seq23</td>
<td>1205</td>
<td>5.39Å</td>
<td>0.71</td>
</tr>
<tr>
<td>Seq34</td>
<td>950</td>
<td>5.37Å</td>
<td>0.66</td>
</tr>
<tr>
<td>Seq45</td>
<td>977</td>
<td>5.37Å</td>
<td>0.68</td>
</tr>
<tr>
<td>Seq05</td>
<td>5241</td>
<td>4.74Å</td>
<td>0.71</td>
</tr>
<tr>
<td><strong>Low dose</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Avg. x2</td>
<td>4528</td>
<td>4.96Å</td>
<td>0.66</td>
</tr>
<tr>
<td>Avg. x5</td>
<td>1718</td>
<td>4.77Å</td>
<td>0.63</td>
</tr>
<tr>
<td>Avg. x10</td>
<td>1046</td>
<td>4.76Å</td>
<td>0.45</td>
</tr>
</tbody>
</table>

Average of 10 consecutive frames, which shows a dramatically low correlation. This value confirms that averaging consecutive frames over one-second long interval deletes structural features, rather than enhancing them. However, the second and worst conclusion following this analysis is the low reliability of the resolution value.

The 3D model of ferritin protein extracted from the image sequence recorded at low dose and processed by averaging 10 consecutive frames is associated to both the lowest resolution and the highest correlation values. The analysis on the electron dose seems not to fully explain the problem of the missing patches, since they appear to be missing also in the case of low dose imaging. Once more, molecular dynamics simulations were performed in order to deeply investigate this phenomenon. Time domain was set at the center of this new analysis. The system under investigation was reconstructed, describing the characteristics of the media, the temperature, and position of the atoms of the ferritin protein in 3D space. Hence, the behaviour of the particles were simulated when immersed in a liquid media. The small conformational changes due to the thermal fluctuations were registered, creating a “dynamical map” of the ferritin protein, which accounts for the different

![Figure 4.53: Superimposition of the x-ray (multi-colours) and cryo-erm (in purple) density maps.](image)
conformations. The whole structure of the ferritin protein obtained by superimposing 60 different simulated conformations is depicted in Figure 4.54. The α-helical units modelling the rigid structure are coloured in orange, while the coils in white.

Figure 4.54: Two different views of the structure of the ferritin protein obtained by superimposing 60 different conformations obtained during the molecular dynamics simulation. The α-helical units composing the rigid structure are depicted in orange.

Figure 4.55: a-c) Schematic representation of the results of the molecular dynamics simulation performed on the structure of the ferritin protein. The simulated models are comprised of the atoms occupying a specific position in space during the associated time interval, measured in nanoseconds. The models are shown from two different point of views, capturing the profile on the x-z plane (on the top row), and on the x-y plane (on the bottom row).

In this fashion, a 4D matrix is associated to the simulated structure, tracking the trajectories of the atoms modelling the structure of the ferritin protein. This 4D matrix is named residence time. In Figure 4.55 the results of the dynamic simulation are reported. The
simulation has been run for 5\( \mu \)s, extracting data every 20ps. The simulated 3D models are extracted at specific time points, and are composed of the atoms of the ferritin protein occupying a specific point in space for the whole time interval. Thus, the first model (Figure 4.55a) depicts the volume that the ferritin protein can occupy in the first snapshot, the second model all the atoms not moving for 40ns of simulation (Figure 4.55b), the second model all the atoms not moving for 80ns of simulation (Figure 4.55c). The number of atoms resulting from the simulation decreases proportionally to the simulation time. Moreover, this molecular dynamics analysis was used to further investigate the behaviour of the fundamental unit of the ferritin structure. The following data are quantified by applying the root mean squared displacement (RMSD) with respect to the referent atomic positions, and the results are depicted in Figure 4.56. The structure of the ferritin results to be comprised of rigid elements, resulting in the long simulation time model. Consequently, the patches missing from the density maps reconstructed via both BT and BPA may be linked to the dynamic of the ferritin protein, which cannot be captured by the temporal resolution of the LTEM.

In order to validate these results, several 3D maps of ferritin protein were reconstructed both via BPA and BT, by processing the image sequences recorded at high and low dose, described in Table XV and Table XX respectively. Both image sequences were processed by applying N2V, trained on two different sets of 200 different images in order to account for the different imaging conditions. In Figure 4.57 the initial frames of both image sequences are depicted, displaying the original and the denoised version.

After mitigating the noise corrupting the original image sequences, the processed frames were fed in input to Deep LiqID, which extracted a total of 10348 (5341 and 5007 in the two five seconds long sequences) and 7683 different profiles for the high dose and the low dose image sequences, respectively. The particle identification step was performed by setting
Figure 4.57: Comparison of the performances of N2V applied to two different image sequences recorded at high dose (on the left) and at low dose (on the right).

a sensitivity threshold of 0.3. It is worth noticing how Deep LiqID outperforms EMAN2, which has been previously applied to the same image sequences (Figure 4.58).

Figure 4.58: Comparison of the performances of EMAN2 (a) and Deep LiqID (b) applied to an image recorded via LP EM of a sample of ferritin protein.

The 3D reconstructions depicted in Figure 4.45 and Figure 4.51 are comprised of 5241 and 5950 different profiles, respectively. The same coordinates were further used to identify the same profiles in the original image sequences in order to evaluate the effect of N2V on the 3D reconstruction step. The 3D maps obtained via BPA performed by using all the identified profiles are depicted in Figure 4.59.

The 3D maps reconstructed from the image sequence processed via N2V result to be much more detailed than the corresponding 3D models resulting from the original frames. Moreover, it is worth mentioning how the initial 3D models do not appear as simple spheres, conversely to the results previously obtained. These density maps were reconstructed via
BPA by using many more profiles than the density maps in Figure 4.45 and Figure 4.51, due to the application of Deep LiqID. The imaged particles provide the reconstruction algorithm with the same number of profiles, however their contributions varies with respect to their dynamical behaviour. Fixed particles produce a number of copies of the same profiles, while moving particles a wide variety of different profiles. Consequently, the resulting density maps may be biased by the available profiles, similarly to cryo-EM. However, there are no further similarities between the still particles imaged via PL EM, and the ones adsorbed by the air-water interface in cryo-EM. In fact, particles attached to the windows of the liquid cell do not undergo fast denaturation processes, thus providing the reconstruction algorithm only with replica of the same particle profile, and limiting the negative effects on the final reconstruction. Moreover, in order to verify the findings of the dynamics simulation and their weights into the final reconstructions, the full image sequence was split into different time intervals and the corresponding profiles were used to reconstruct different 3D maps via BPA.

Initially, the image sequence recorded at high dose composed of 100 frames was divided into two, five and then ten consecutive not overlapping time intervals. The particle profiles identified in these subsets of frames were further processed to reconstruct the 3D maps shown in Figure 4.60. Similarly, the five second long image sequence recorded in low dose and comprised of 50 frames was split into five intervals, generating five different 3D maps. The results are depicted in Figure 4.61.
Figure 4.60: Schematic representation of the 3D reconstructions obtained via BPA and resulting from the image sequence recorded at high dose (Figure 4.57). The image sequence was processed via N2V and split into different time intervals. 

a) Density maps obtained by splitting the original image sequence into two different 5s long intervals. 

b) Density maps obtained by splitting the original image sequence into five different 2.5s long intervals. 

c) Density maps obtained by splitting the original image sequence into ten different 1s long intervals. Every density map is associated to the number of profiles included in the reconstruction process.

Taking into account the improvements guaranteed by the application of N2V, this analysis was performed only on the denoised image sequences. The 3D maps obtained from the image sequence recorded at high dose show how the details of the structure emerge proportionally to the number of time intervals. The density maps reconstructed from the two five seconds long time intervals (Figure 4.60a1 and Figure 4.60a2) result to be similar, characterised by a wide missing patch and a generally smooth surface. Theses 3D models result from the averaging operation of over 5000 different profiles, extracted from the observation of circa 120 particles moving over 10 seconds. The difference in the roughness of the surface of the two models may be a consequence of the long time exposure of the particles to the beam. Consequently, BPA erases the structural information, but creates an average reconstruction modelling the possible conformations that the particles of ferritin protein can assume over time. Conversely, limiting the observation time and consequently the number of profiles concurring to the 3D map highlights the elements composing the
Figure 4.61: Schematic representation of the 3D reconstructions resulting from the image sequence recorded at low dose. The 3D maps in purple correspond to the different time interval composing the image sequence. Every 3D model is associated to the number of profiles included in the reconstruction process.

structure of the ferritin protein. Short time intervals pool together different profiles extracted from particles undergoing similar changes. Comparable results can be derived by observing the results depicted in Figure 4.61.

It is important to highlight the differences shown by the reconstruction process when applied to N2V processed image sequences. In particular, the 3D models depicted in Figure 4.51 and Figure 4.61 differ only for the application of N2V. The first models appear dusty, and they need an additional averaging step in order to detect any structural feature in the 3D models.

Figure 4.62: Schematic representation of the results of the BT analysis performed on the image sequence recorded at high dose. On the left, the last frame extracted from the image sequence. The blue dotted circles labels the particles chosen to reconstruct, while the red dots marks the initial positions. On the right, the corresponding five 3D models, obtained via BT.

Additionally, in order to further investigate the variations on the structures of the imaged ferritin protein particles, BT analysis was performed on both the image sequences
at high and low dose depicted in Figure 4.57. The results of this analysis are reported in Figure 4.62 and Figure 4.63, respectively. Both figures show the last frame of each image sequence, together with the five particles selected to be reconstructed. The light blue dots mark the corresponding positions in the first frame of the sequence. In order to acquire as many profiles as possible, the particles have been chosen amongst the ones moving the most. Generally, the particles imaged at low dose seem to have a reduced movement, showing negligible variations of the locations. The 3D models derived from the image sequence recorded at high dose resulted well defined, clearly showing the helices of the ferritin protein. The density maps reconstructed from the profiles produced by $p_1$, $p_2$, $p_3$, $p_4$ are surrounded by dust, due to the noise in the image. On the contrary, the 3D map associated to the fifth selected particle reaches the highest quality, showing only the rigid cage-like structure. Particle $p_5$ covers a remarkable portion of space over the observation time, thus having the camera capture the evolution over time of the particle. In this fashion, the BT technique produced a density map that matches the results obtained by the molecular dynamics simulation at long residence time. Conversely, the results depicted in Figure 4.63 are more difficult to understand. The low resolution and number of profiles used to perform the BT analysis lower the quality of the 3D maps. The produced volumes are incomplete, presenting a high number of missing patches, mainly derived from the small movements experienced by the particles, and the low contrast. All the density

![Figure 4.63: Schematic representation of the results of the BT analysis performed on the image sequence recorded at low dose. On the left, the last frame extracted from the image sequence. The blue dotted circles labels the particles chosen to reconstruct, while the red dots marks the initial positions. On the right, the corresponding five 3D models, obtained via BT.](image)
maps in Figure 4.62 and Figure 4.63 were filtered in order to reduce the surrounding dust, by removing all the elements with a diameter smaller than a few angstroms.

The results presented in this section were obtained while this thesis was being written, and the investigation can not be considered complete. Hence, the molecular dynamics simulation investigating the similarity between the reference reconstruction obtained via x-ray tomography and the 3D models reported in Figure 4.59, Figure 4.60, Figure 4.61, Figure 4.62, and Figure 4.63 could not be performed.

It is worth mentioning that at the same time this molecular dynamics analysis was implemented, a slightly different approach to perform BT and BPA was proposed. This second strategy exploits the \textit{a priori} known symmetry characterising the structures of the specimen to reconstruct, and replaces the coarse model with the dynamics maps resulting from simulations. Despite this method is still under investigation, a preview of the performance can be appreciated in Figure 4.64.

\textbf{Figure 4.64: Schematic representation of the results of both BT and BPA performed on the image sequence recorded at high dose.} The model obtained by BT is compared with the x-ray tomography reconstruction; while the BPA model is compared with the result of the dynamics simulation shown in Figure 4.54. For both BT and BPA, from left to right the refined models, the comparison of two adjacent halves, and the comparison of the cross-sections.
The results are obtained by processing the image sequence recorded in high dose. The 3D map resulting from BT matches in shape and size the model obtained via x-ray tomography. Moreover, the hole resulting from the junctions of the α-helices enables the alignments of the models, stressing the quality of the results. Conversely, BPA averages the contributions of more than a hundred particles, representing different conformations. Interestingly, the BPA map matches the result of the simulation (Figure 4.54), precisely overlapping the α-helices arrangements.

Moreover, in order to further investigate the role of dynamics maps in the reconstruction process, the workflow leading to the results in Figure 4.64 was modified by firstly removing any constraints from the initial model, and secondly replacing the dynamics map set with the density map deriving from the x-ray reconstruction. The results of these two methods are reported in Figure 4.65.

![Figure 4.65](image)

*Figure 4.65: Schematic representation of the results of both BT and BPA performed on the image sequence recorded at high dose. The workflow producing the results in Figure 4.64 was firstly modified by removing the dynamics map as initial model, and secondly by replacing the dynamics map set as initial model with the density map reconstructed from the x-ray structure.*

The generated structures show that templating the refinement improves the resolutions considerably for both the BT and BPA reconstruction methods, capturing the core-shell structure of ferritin protein. However, only the BT maps show the expected octahedral symmetry, while the BPA maps show a spherical structure with not distinguishable features. The lack of details is very likely due to the fact proteins can explore wider conformational spaces in liquid than in vitrified water and at room temperature.

The results produced by the workflow using the dynamics map as initial model are promising, producing high quality results in the reconstruction of specimens imaged via
LP EM. However, the results shown in this thesis are obtained at early stages of the development of this approach. Further analysis are required and will be performed in the next future.
This thesis describes the achievements reached during a research project lasting almost four years. It is based on LP EM, a well known technique still in its early stages at the time this project started. In the last decade, the interest in LP EM and its applications grew exponentially, with more and more researchers from different fields trying to disclose its full potential, and discover its limitations. The advent of LP EM represented a great possibility to observe and study specimens in their liquid native state over time. The original main objective of this thesis consisted on the development of a method to reconstruct the 3D volume of soft organic biomolecules at the nanoscale surrounded by liquid media. Unfortunately, the path leading to the birth of this novel method was complicated by unseen elements, never studied before. In fact, the presence of the liquid revolutionised the field of microscopy. It represents an additional source of noise compromising the quality of the produced images. Moreover, it allows the particles to be displaced not only on a single plane, but actually to move perpendicularly to the imaging plane, i.e., close and away from the detector. Consequently, focusing the right objects becomes a very important task of the whole imaging process.

Therefore, a significant part of this research project was dedicated to the characterisation of the features of LP EM, understanding and quantifying how the presence of the liquid affects the imaging process. Initially different denoising methods were developed, trying to improve the quality of the images, and keeping the associated processing time low. Balancing the goodness of the computer vision approaches and their execution time represented the leitmotif of this thesis. The research of this trade-off motivates the progressive replacement of the initial traditional algorithms with deep learning-based approaches. Deep learning is a relatively new way to develop software able perform complicate tasks in very short amount of time. In particular, this path was chosen to process the images recorded via LP EM.

The first proposed method concerns the reduction of the noise corrupting the images, in order to improve the quality and the resolution of the images. Processing data heavily corrupted by noise could lead to partial or totally inaccurate results. Conventional de-
noising methods are tailored on the noise distribution, require the clean-noisy pair images, and do not account for the presence of the liquid. Consequently, conventional methods do not perform well on LP EM micrographs. N2V is a deep learning-based method that suppresses the noise without the need of the corresponding noiseless image. This feature is of great importance as in the case of EM the noiseless images are never available. However, this method has to be trained on hundreds of images similar to the ones to denoise. Unfortunately, a standard to image specimens via LP EM is hard to define, as the imaging conditions are specifically tuned on the specimen to image. In details, the nature of the specimen affects the focus value, the magnification, and then the pixel size. Nevertheless, N2V may play a very important role in the future of LP EM, as it was proven to be fast and robust, producing high quality results in many different imaging conditions. N2V significantly mitigates the effect of the noise corrupting LP EM images, consequently improving the resolution of the recorded micrographs, and allowing the detector to capture the features of the specimens, otherwise covered by noise. In this fashion, the bottleneck of the resolution is loosened, thus enabling a plethora of different analyses requiring high resolution images.

After mitigating the effect of the noise, the focus of the research moved to the identification of the particles in the images. This task is complicated by the different heights the particles can occupy in the sample holder. Objects far from the detector are difficult to identify as are characterised by low contrast, show few details, and are generally smaller than the ones on the focus plane. Consequently, methods based on the development of a template require a very detailed and wide library of instances, and are thus very hard to implement. Conversely, Deep LiqID was trained to identify nanoparticles of different shapes and natures, and occupying different heights in the holder; thus learning the features defining the objects in liquid and distinguishing them from the features of the liquid. Deep LiqID was proven to outperform EMAN2 - one of the most used tools for particle picking -, especially in the case of images characterised by low contrast, where the identification task is hard even for human operator. Moreover, Deep LiqID is fully automated, does not require any human input, and its performance can be still improved, by enriching the dataset with new labels registered in different imaging conditions, and re-train the deep artificial network. Having an object identification tool enables many
different analyses, such as tracking the motion of the objects over time, or measuring their physical properties.

All the developed techniques had the dual effect of investigating the potential and the limitations of LP EM, and at the same time improving its performance, consequently enabling novel analyses. What clearly emerged was that the presence of the liquid media enables the observation of the dynamics of organic nanoparticles in their native state, but severely limits the resolution and the quality of the recorded images. Thus, LP EM cannot represent a competitor to cryo-EM, which produces the best results in terms of quality and resolution, producing sharp and clear images. Similarly, 3D reconstruction techniques based on LP EM and cryo-EM cannot produce comparable results, with the former offering no atom level information, and the latter producing 3D models with resolution values several times lower. Nevertheless, the loss in spatial resolution is compensated by the access to dynamical resolution. BPA averages the profiles extracted from a population of particles, a process that blurs structural information, but captures all the different configurations assumed by the imaged particles. Most importantly, every particle evolves independently, stochastically, following a Brownian dynamics, as well as being free to explore different conformational arrangements. Conventionally, such dynamics information cannot be accessed, as cryo-EM or x-ray crystallography need to vitrify the water and crystallise the proteins, respectively. Moreover, BT coupled with molecular dynamics simulation accesses structural information by extracting the profiles from a single object, while the best cryoEM reconstructions requires about $10^6$ independent particles.

BPA and BT represents the final goal of this research project. However, the designing process of these reconstruction methods was not linear, as new fields of investigation and application raised at every step. Answering to these side paths enriched the main research topic, bringing more and more awareness of the potential of the LP EM. Computer vision was the key to this progress by providing the necessary tools to push further the limits of knowledge. Most of these tools were borrowed from different fields of application, showing the importance of cross fertilisation for the research. Moreover, all the methods composing this work were designed in a modular way, being developed independently, but defining a single framework. In this fashion, improving the performance of a single method will affect the reconstruction results. In particular, the power of learning-based methods relies in their flexibility, and the inner ability of being continuously updated.
The performance reached by both N2V and Deep LiqID can improve by simply bringing in new data and information. Producing images with high SNR and identifying highly defined nanoparticles will enhance the quality of the final BPA and BT reconstructions, thus enabling detailed analysis on the dynamics of organic nanoparticles. Therefore, the development of BPA and BT paved the way to a completely new frontier in structural biology, expanding functional studies towards dynamics processes.

The last results presented at the end of Chapter 4 shed light on the plans for the next future, highlighting the central role of molecular dynamics. Pairing molecular dynamics to the latest results of CV may have a significant impact in the progress of the field of LP EM. In particular, dynamics analyses are limited by the acquisition rate of the camera (i.e. in our specific case), which hinders the observation of phenomena faster than 100ms. Thus, the development of a novel method improving the quality of the micrographs recorded at more than 10fps would loosen this bottleneck. However, learning-based methods may be developed to improve not only temporal resolution, but also spatial resolution. In fact, CNNs may be applied to perform the so-called “super resolution”, a class of CV methods widely investigated in the last few years. Moreover, the methodology proposed in this thesis may represent the starting point of an ambitious project, which may include the element of time into the developed methods. For instance, N2V may be implemented by using Recurrent Neural Networks, architectures able to extract features from temporal sequences, rather than single elements. This class of ANNs may be also used to analyse the behaviour of the observed specimens, classifying dynamics events or identifying anomalies. An obstacle to the realisation of similar methods relies in the absence of ground truth and in the difficulty of producing high quality labels. However, the recent progress in the field of CV are pushing the frontiers of semi-supervised and unsupervised learning further and further, strategies that may result of great relevance for the future of these projects.

Unfortunately, the ideas proposed in this last sections are only speculations, as the lack of time pulled a break to the investigations conducted in this research project, which hence cannot be considered finished. However, the work presented in this thesis set the basis for future investigations, opening new scenarios of analysis, and planting the seeds for future progress in the fields of LP EM and CV.
In this thesis, several computer vision methods were proposed in order to improve the performance of the LP EM technique, and generate dynamics 3D model of the imaged nanoparticles. Despite the discussion focuses mainly on the design, the performance, and the results of each method, the role and the importance of the organic samples in this research work cannot be neglected. It is worth mentioning that the sample preparation and the imaging process were not performed by the author of this thesis, but were realised by an external collaborator.

In this appendix a description of the techniques adopted to prepare the samples appearing in this thesis is provided.

*Equine spleen apoferritin*
Equine spleen apoferritin was purchased from Sigma-Aldrich and diluted at 2mg/mL in PBS.
Ferritin nanoparticles were used to train N2V (*Section 3.2.3*), Deep LiqID (*Section 3.3.3*), and perform BPA and BT reconstruction analysis (*Section 3.5*).

*PMPC-PDPA*
PMPC-PDPA amphiphilic block copolymers were synthesised by atom transfer radical polymerisation [294]. PMPC-PDPA particles were formed by the pH-switch method. First, the polymers were dissolved in PBS at pH 2 in a concentration of 10mg/mL solution; then, the pH was slowly raised to 7.4 adding sterile NaOH (0.5M) dropwise. The initial transparent solution became milky due to the formation of the polymer nanostructures. The solution was purified by size-exclusion chromatography (SEC), where the solution was passed through a size exclusion column filled with Sepharose 4B, bead diameter between 45 and 165 µm, purchased from Sigma-Aldrich. SEC was performed in order to remove polymer aggregates and larger structures. The PMPC-PDPA polymer particle solution was diluted to 1.5mg/mL in PBS and imaged.
PMPC-PDPA structures were used to evaluate the performance of the denoising pipeline (Section 3.2.1), train Deep LiqID (Section 3.3.3), and to perform BT reconstruction analysis (Section 3.5).

**PEG-PMET**

Block copolymer PEG-PMET vesicles were produced by the solvent-switch method. Briefly, PEG-PMET (molecular weight (Mw): 25kDa by 1H-NMR (nuclear magnetic resonance), 26kDa and 1.4 polydispersity by gel permeation chromatography (GPC)) was previously synthesised according to Yamada et al. [295], with some modifications. Then, 10mg of the isolated polymer were dissolved in 0.5mL of a mixture 1:1 of tetrahydrofuran:dimethyl sulfoxide (THF:DMSO). Next, 1.15mL of milliQ water was added using a syringe pump at 1µL/min rate, while keeping the sample under stirring at 500rpm at room temperature. Finally, the sample was diluted with 1.35mL of milliQ water and dialysed against water (×5) using Spectra/Por 6 Dialysis Tubing, 3.5kD MWCO (Spectrum labs). The polymer vesicle solution was further diluted to 1.5mg/mL in PBS and imaged. PEG-PMET structures were used to evaluate the performance of the denoising pipeline (Section 3.2.1).

**PEG-PDPA**

PEG-PDPA and copolymers were synthesised as reported by the atom-transfer radical polymerisation method [296]. PEG-PDPA structures were used to evaluate the performance of the identification in dry state (Section 3.3.4).

**SiO₂**

For the synthesis of smooth silica, a modified Stöber method was used with tetraethyl orthosilicate (TEOS) as silica precursor and aqueous ammonium as catalyst for the reaction. In a typical reaction, a solution of 10mL ethanol (EtOH) with 3mL of deionized water and 0.8mL of aqueous ammonia was prepared. After 30 minutes, 2mL of TEOS was dropwise added to the previously prepared solution and left stirring for 12 hours. The synthesized particles are collected by centrifuging at 12Krpm for 5min and the samples are washed (x3) with deionized water and (x3) with EtOH.
SiO$_2$ nanoparticles were used to train Deep LiqID (*Section 3.3.3*), and perform identification in liquid state (*Section 4.4.1, Section 3.3.2, and Section 3.3.3*).


39. Dieter Typke, Christopher J. Gilpin, Kenneth H. Downing, and Robert M. Glaeser. Stroboscopic image capture: Reducing the dose per frame by a factor of 30
CITED LITERATURE (continued)


47. J. Ryan Feathers, Katherine A. Spoth, and J. Christopher Fromme. Surpassing the physical nyquist limit to produce super-resolution cryo-EM reconstructions. jun 2019.


125. D.S. Goodsell. Human papillomavirus and vaccines. RCSB Protein Data Bank, may 2018.


CITED LITERATURE (continued)


234. Antonia Creswell, Kai Arulkumaran, and Anil A. Bharath. On denoising autoencoders trained to minimise binary cross-entropy.


272. Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hier-
archies for accurate object detection and semantic segmentation.


275. Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal
speed and accuracy of object detection.

276. Guoxu Liu, Joseph Christian Nouaze, Philippe Lyonel Touko Mbouembe, and Jae Ho
Kim. YOLO-tomato: A robust algorithm for tomato detection based on

277. Weijun Chen, Hongbo Huang, Shuai Peng, Changsheng Zhou, and Cuiping Zhang,
YOLO-face: a real-time face detector. The Visual Computer, 37(4):805–813,
mar 2020.

278. John Canny. A computational approach to edge detection. IEEE Transactions on

279. Meng Liu, Azzurra Apriceno, Miguel Sipin, Edoardo Scarpa, Laura Rodriguez-
Arco, Alessandro Poma, Gabriele Marchello, Giuseppe Battaglia, and Stefano
Angioletti-Uberti. Combinatorial entropy behaviour leads to range selective
binding in ligand-receptor interactions. Nature Communications, 11(1), sep
2020.

280. Z. Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment:
From error visibility to structural similarity. IEEE Transactions on Image

281. Berend Smit Daan Frenkel. Understanding Molecular Simulation. Elsevier LTD,

282. Alexis Rohou and Nikolaus Grigorieff. CTFFIND4: Fast and accurate defocus esti-
mation from electron micrographs. Journal of Structural Biology, 192(2):216–
221, nov 2015.

283. E C Theil. Ferritin: Structure, gene regulation, and cellular function in animals,
plants, and microorganisms. Annual Review of Biochemistry, 56(1):289–315,
jun 1987.

284. Canhui Wang, Qiao Qiao, Tolou Shokuhfar, and Robert F. Klie. High-resolution
electron microscopy and spectroscopy of ferritin in biocompatible graphene
liquid cells and graphene sandwiches. Advanced Materials, 26(21):3410–3414,
feb 2014.


295. Shuhei Yamada, Koichi Koga, Atsushi Sudo, Mitsuaki Goto, and Takeshi Endo. Phosgene-free synthesis of polypeptides: Useful synthesis for hydrophobic polypeptides through polycondensation of activated urethane derivatives of