Incremental Refinement of Image Salient-point Detection

Yiannis Andreopoulos* and Ioannis Patras

ABSTRACT

Low-level image analysis systems typically detect "points of interest", i.e. areas of natural images that contain corners or edges. Most of the robust and computationally efficient detectors proposed for this task use the autocorrelation matrix of the localized image derivatives. Although the performance of such detectors and their suitability for particular applications has been studied in relevant literature, their behavior under limited input source (image) precision or limited computational or energy resources is largely unknown. All existing frameworks assume that the input image is readily available for processing and that sufficient computational and energy resources exist for the completion of the result. Nevertheless, recent advances in incremental image sensors or compressed sensing, as well as the demand for low-complexity scene analysis in sensor networks now challenge these assumptions.

In this paper, we investigate an approach to compute salient points of images *incrementally*, i.e. the salient point detector can operate with a coarsely-quantized input image representation and successively *refine* the result (the derived salient points) as the image precision is successively refined by the sensor. This has the advantage that the image sensing and the salient point detection can be terminated at any input image precision (e.g. bound set by the sensory equipment or by computation, or by the salient point accuracy required by the application) and the obtained salient points under this precision are readily available. We focus on the popular detector proposed by Harris and Stephens and demonstrate how such an approach can operate when the image samples are refined in a bitwise manner, i.e. the image bitplanes are received one-by-one from the image sensor. We estimate the required energy for image sensing as well as the computation required for the salient point detection based on stochastic source modeling. The computation and energy required by the proposed incremental refinement approach is compared against the conventional salient-point detector realization that operates directly on each source precision and cannot refine the result. Our experiments demonstrate the feasibility of incremental approaches for salient point detection in various classes of natural images. In addition, a first comparison between the results obtained by the intermediate detectors is presented and a novel application for adaptive low-energy image sensing based on points of saliency is presented.

Index Terms— *salient point detection, incremental refinement of computation, low-level feature detection, progressive image sensing*

EDICS CODE: SRE-LOWR, FLT-OTHR

Yiannis Andreopoulos is with the University College London, Dept. of Electronic & Electrical Engineering, Torrington Place, WC1E 7JE, London, UK; Tel: +44-20-76797303; fax: +44-20-73889325; email: iandreop@ee.ucl.ac.uk.

Ioannis Patras is with the Queen Mary University of London, Dept. of Electronic Engineering, Mile End Road, E1 4NS, London, UK; Tel: +44-20-7882-5345; fax: +44-20-7882-7997; e-mail: i.patras@elec.qmul.ac.uk.

*Corresponding author. Address and contact information: see above.

Parts of this paper were presented in the 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Y. Andreopoulos acknowledges the support of the EPSRC, grant EP/F020015/1.

I. INTRODUCTION

Low-level feature detectors attempt to isolate image areas that contain visually important data, such as edges or corners. Many such approaches have been proposed within the last 30 years [1]-[3] and they are generally termed as detectors of points of interest or salient point detectors (SPDs). Some approaches are stand-alone systems for low-level image analysis [4]; others feed the results into systems performing higher-level image understanding tasks [5]. Comparative tests carried out amongst different approaches demonstrated that detectors that combine the smoothed image derivates via the autocorrelation matrix are robust and achieve some of the best results [6]. Perhaps the most prominent candidate in this category is the edge and corner detector proposed by Harris and Stephens [2].

Initial research in the area (e.g. Moravec’s work) begun because low-level real-time image analysis is extremely useful for various applications in robotics [1], real-time surveillance and monitoring [7], etc. Computational and sensing energy requirements are very important for these applications. Today, even though computing systems have evolved to the extent where derivative-based approaches can be executed fast, the image processing needs have also increased dramatically since image resolutions are higher and there is a significant demand for processing high frame-rate videos or images and views derived from multiple cameras in systems with limited computational and energy resources, e.g. in video sensor networks [8]. Hence, the required computational and energy resources remain an important concern. Finally, most modern realization platforms utilize dynamic task scheduling [9], and low-power task scheduling, e.g. via the use of dynamic voltage scaling [10]. Hence, multimedia applications need to be able to produce the best-possible result under rapidly-changing system resources [10].

Besides the computational aspects, new trends have also emerged in the image sensor arena. Approaches for compressed sensing [11] are assuming the use of limited sensory equipment (even up to single-pixel sensors [12]) to derive a resolution and quality-refinable approximation of the input visual data. A more straightforward and already fully-functional approach is based on CMOS image sensors capturing the input source incrementally from the most-significant bits (MSBs) to the least-significant bits (LSBs) using successive analog-to-digital conversion [13].

In this paper, we are considering the derivative-based detector of Harris and Stevens and extend it to support incremental derivation of image salient points with increased input-image precision. This complies with the incremental CMOS-based image sensing approach proposed by Yang et al [13] that produces image bitplanes hierarchically, from the MSBs to the LSBs, as shown in Figure 1. We reformulate the detector for incremental refinement and thereby create a hierarchical computation framework for the derivation of image salient points for each new input image bitplane. We exploit the fact that the salient points of the previous input are known when new input bitplanes are processed and localize the computation and image sensing in a window surrounding the position of each previously-found salient point. This is expected to reduce the energy and computational requirements in comparison to the straightforward approach that senses the entire image and then computes the salient points. We investigate the computational requirements for a variety of input images in comparison to the conventional approach and perform a detailed study to demonstrate the conditions under which the processing and sensing power of the proposed approach is reduced versus the conventional approach that is not refinable and requires sensing and processing the entire input image.

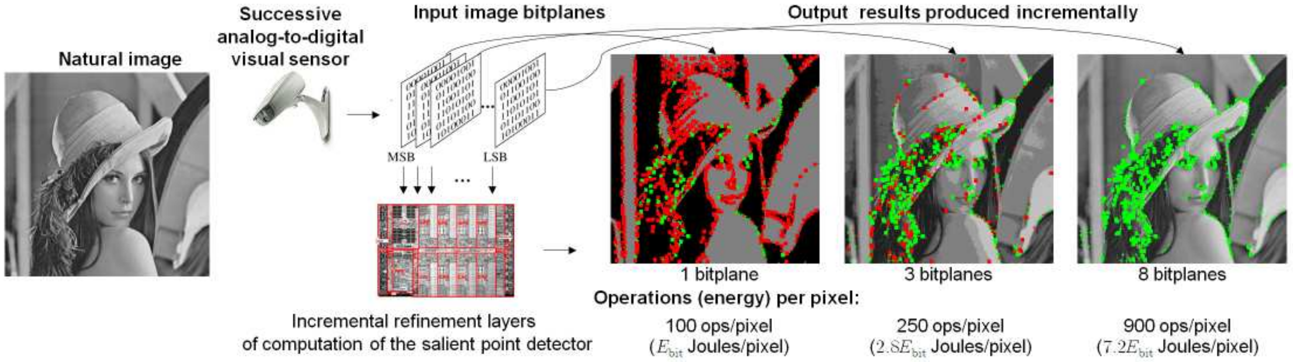

Natural image
Successive analog-to-digital visual sensor
Input image bitplanes
Output results produced incrementally
MSB
LSB
Incremental refinement layers of computation of the salient point detector
1 bitplane
3 bitplanes
8 bitplanes
Operations (energy) per pixel:
100 ops/pixel (E_{bit} Joules/pixel)
250 ops/pixel ($2.8E_{\text{bit}}$ Joules/pixel)
900 ops/pixel ($7.2E_{\text{bit}}$ Joules/pixel)

Figure 1. Overall framework for successive bitplane-based image sensing and incremental salient-point detection. The visual sensor captures the input source incrementally from the most-significant bits (MSBs) to the least-significant bits (LSBs). The output results are produced incrementally from MSB to LSB. Incremental production of the results leads to progressive improvement of quality under increased computation (operations per pixel) and increased sensing energy (in multiples of the basic sensing energy per bit – E_{bit}).

This paper is structured as follows. Section II presents the proposed formulation enabling incremental refinement for the computation of image salient points. Section III analyzes the computational and energy requirements for incremental salient-point detection in comparison to the conventional approach where the salient point detector (SPD) operates under fixed source precision. Section IV presents experiments deriving the performance of the incremental detector versus the conventional approach. Several aspects and possible extensions are discussed and a new application of salient-point driven image sensing is presented. Finally, Section V draws our conclusions.

II. FORMULATION OF INCREMENTAL REFINEMENT OF SALIENT POINT DETECTION

A. Salient Point Matrix Derived using the Harris Algorithm

The basic algorithm of Harris and Stephens [2] for image salient-point detection begins by calculating an approximation for the horizontal and vertical first-order derivatives of the input image. For an image $\mathbf{I}$ consisting of $R \times C$ pixels, if we define the difference matrix:

$$\mathbf{D} = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix},$$

the approximation of the horizontal and vertical first-order derivatives of the image is:

$$\mathbf{X} = \mathbf{I} * \mathbf{D}\,,\ \mathbf{Y} = \mathbf{I} * \mathbf{D}^T\,, \tag{1}$$

where operator $*$ denotes two-dimensional (2D) convolution.

In order to reduce the influence of noise, smoothing with a Gaussian window is used [2]:

$$\mathbf{A} = (\mathbf{X} \circ \mathbf{X}) * \mathbf{G} \tag{2}$$

$$\mathbf{B} = (\mathbf{Y} \circ \mathbf{Y}) * \mathbf{G} \tag{3}$$

$$\mathbf{C} = (\mathbf{X} \circ \mathbf{Y}) * \mathbf{G} \tag{4}$$

where operator $\circ$ denotes the Hadamard (element-wise) matrix product and $\mathbf{G}$ denotes the two-dimensional Gaussian window formed by:

$$\mathbf{G}[r,c] = \frac{1}{f_{\mathbf{G}}} \exp\left[(-r^2 - c^2)/2\sigma^2\right] \tag{5}$$

with $f_{\mathbf{G}}$ the normalization factor that ensures that the 2D Gaussian filter has unitary gain. The typical size of the window is: $0 \le r, c < 6\lfloor \sigma^2 \rfloor$. The autocorrelation matrix of the image derivatives can be defined as the 2×2 matrix:

$$\mathbf{M} = \begin{bmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C} & \mathbf{B} \end{bmatrix} \tag{6}$$

with elements defined in (2)-(4). In the salient-point detection process, following the original paper of Harris and Stephens [2], we use the “trace matrix” of $\mathbf{M}$, as well as its “determinant matrix”:

$$\mathrm{Tr}(\mathbf{M}) = \mathbf{A} + \mathbf{B} \tag{7}$$

$$\det(\mathbf{M}) = (\mathbf{A} \circ \mathbf{B}) - (\mathbf{C} \circ \mathbf{C}). \tag{8}$$

The formulation proposed by Harris and Stephens for the salient point matrix is:

$$\mathbf{R} = \det(\mathbf{M}) - k \cdot \mathrm{Tr}(\mathbf{M}) \circ \mathrm{Tr}(\mathbf{M}), \tag{9}$$

with k a scaling factor which is a parameter of the detection algorithm. The second term of (9) is used to eliminate contour points with one strong eigenvalue in $\mathbf{M}$ [6]. Positive values of $\mathbf{R}$ occur in the corner regions, negative values in the edge regions, and small values in the flat regions. Consequently, thresholding $\mathbf{R}$ gives the areas containing salient points.

B. Selection of Salient Points using $\mathbf{R}$

There exists a variety of methods to derive the location of salient points, depending on the application of interest. In this paper we follow the common approach [2] [6] of comparing the local maximum and the local minimum of $\mathbf{R}$ (within a neighborhood of 3×3 pixels) with a certain threshold Θ. The local maxima of $\mathbf{R}$ that are larger than Θ are labeled “corner” points. The local minima that are smaller than $-\Theta$ are labeled “edge” points. Typically, Θ is set to a certain percentage θ of the maximum observed value of $\mathbf{R}$ [6]. In this way we end up with a disjoint set of salient points, as shown in the illustration examples of Figure 1. The junction completion algorithm that merges the derived salient points (e.g. in order to derive continuous edge profiles from the salient points representing edges) can be a separate part of the system depending on the application of interest [2]-[5]. Hence, any further processing of the salient points, such as extraction of descriptors for higher-level reasoning or linkage for edge detection, is beyond the scope of the current paper.

C. Incremental Derivation of the Salient Point Matrix under Increased Image Sensing Accuracy

We now consider the computation of the algorithm of the previous section under the assumption that the image is sensed incrementally, i.e. from the most-significant to the least-significant bitplane, as seen in the framework of Figure 1. There, the image is sensed in a bitplane-by-bitplane fashion, which means that each pixel of image $\mathbf{I}$ is represented by:

$$i_{\text{sensed}}^{0}[r,c] = (-1)^{i_{\text{bit}}^{N}[r,c]} \sum_{n=0}^{N-1} i_{\text{bit}}^{n}[r,c] \cdot 2^{n} \qquad (10)$$

where $i_{\text{bit}}^{n}[r,c]$ is the n th bit of sensed pixel $i_{\text{sensed}}^{0}[r,c]$, with $i_{\text{bit}}^{0}[r,c]$ being the LSB, $i_{\text{bit}}^{N-1}[r,c]$ the MSB, and $i_{\text{bit}}^{N}[r,c]$ reserved for the sign bit (zero for positive values, one for negative values). For typical 8-bit grayscale $R \times C$ -pixel images we have $N = 8$ and for every $0 \le r < R$, $0 \le c < C$: $i_{\text{bit}}^{8}[r,c] = 0$, i.e. all pixel values are positive. In the following part of this section, we describe the steps for the incremental computation of the salient point detector of Subsection II.A assuming that we are operating on any bitplane n , from $n = N-1$ to $n = 0$.

We begin by extending the notation presented above to cover the remaining parts of the SPD. This is performed by the following definitions.

Definition 1: For any quantity a used in the SPD algorithm, a_{sensed}^{n} , $0 \le n \le N-1$, is the computed value of a when the input image $\mathbf{I}$ is sensed from bitplane $N-1$ up to (and including) bitplane n . □

Definition 2: For any quantity a used in the SPD algorithm, a_{bit}^{n} , $0 \le n \le N-1$, is the computed value of a when using only bitplane n of the input image $\mathbf{I}$. □

The notational conventions of Definition 1 and Definition 2 are extended for matrices, e.g. $\mathbf{A}_{\text{bit}}^{n}$ is the matrix containing the computed coefficients of $\mathbf{A}$ when only bitplane n of the input image is used. In the following we derive an algorithm for incrementing the computation of the final result of the SPD (salient point matrix) when the image is sensed bitplane-by-bitplane starting with the most-significant bits.

Starting with the computations for the image derivatives seen in (1), we notice that they can be performed separately for each new bitplane $\mathbf{I}_{\text{bit}}^{n}$ since the multiply-accumulate process performed during convolution is a linear operation that can be broken into separate multiply-accumulate processes for each bit that are summed afterwards. Hence, for input image bitplane $\mathbf{I}_{\text{bit}}^{n}$ we obtain $\mathbf{X}_{\text{sensed}}^{n} = \left(\mathbf{I}_{\text{sensed}}^{n+1} + \mathbf{I}_{\text{bit}}^{n}\right) * \mathbf{D}$. This can be expressed as $\mathbf{X}_{\text{sensed}}^{n} = \mathbf{X}_{\text{sensed}}^{n+1} + \mathbf{X}_{\text{bit}}^{n}$ with:

$$\mathbf{X}_{\text{bit}}^{n} = \mathbf{I}_{\text{bit}}^{n} * \mathbf{D} \, . \qquad (11)$$

Similarly, we obtain $\mathbf{Y}_{\text{sensed}}^{n} = \mathbf{Y}_{\text{sensed}}^{n+1} + \mathbf{Y}_{\text{bit}}^{n}$, with:

$$\mathbf{Y}_{\text{bit}}^{n} = \mathbf{I}_{\text{bit}}^{n} * \mathbf{D}^{T} \, . \qquad (12)$$

In (11), (12), $\mathbf{X}_{\mathrm{bit}}^{n}$, $\mathbf{Y}_{\mathrm{bit}}^{n}$ contain the output of bitplane n. We remark that these matrices will have slightly increased dynamic range in comparison to $\mathbf{I}_{\mathrm{bit}}^{n}$, i.e. they will not contain only binary values, due to the accumulation performed in the convolution operations.

The next step involves filtering with the Gaussian window, as shown in (2)-(4). Due to the non-linearity introduced by the Hadamard products prior to the convolution operations, we break the computation as follows:

$$\mathbf{A}_{\mathrm{sensed}}^{n} = \left[\left(\mathbf{X}_{\mathrm{sensed}}^{n+1} + \mathbf{X}_{\mathrm{bit}}^{n}\right) \circ \left(\mathbf{X}_{\mathrm{sensed}}^{n+1} + \mathbf{X}_{\mathrm{bit}}^{n}\right)\right] * \mathbf{G} \tag{13}$$

which, by expanding the Hadamard product, can be written as:

$$\begin{aligned} \mathbf{A}_{\mathrm{sensed}}^{n} &= \left(\mathbf{X}_{\mathrm{sensed}}^{n+1} \circ \mathbf{X}_{\mathrm{sensed}}^{n+1}\right) * \mathbf{G} + \left(\mathbf{X}_{\mathrm{bit}}^{n} \circ \mathbf{X}_{\mathrm{bit}}^{n}\right) * \mathbf{G} + 2\left(\mathbf{X}_{\mathrm{sensed}}^{n+1} \circ \mathbf{X}_{\mathrm{bit}}^{n}\right) * \mathbf{G} \\ &= \mathbf{A}_{\mathrm{sensed}}^{n+1} + \mathbf{A}_{\mathrm{bit}}^{n} \end{aligned} \tag{14}$$

with:

$$\mathbf{A}_{\mathrm{bit}}^{n} = \left(\mathbf{X}_{\mathrm{bit}}^{n} \circ \mathbf{X}_{\mathrm{bit}}^{n}\right) * \mathbf{G} + 2\left(\mathbf{X}_{\mathrm{sensed}}^{n+1} \circ \mathbf{X}_{\mathrm{bit}}^{n}\right) * \mathbf{G}\,. \tag{15}$$

The last equation shows that the non-linearity introduces the additional term $2\left(\mathbf{X}_{\mathrm{sensed}}^{n+1} \circ \mathbf{X}_{\mathrm{bit}}^{n}\right) * \mathbf{G}$ in the increment of the computation for $\mathbf{A}_{\mathrm{sensed}}^{n}$. In the same manner, we derive the increment for the computation of $\mathbf{B}_{\mathrm{sensed}}^{n}$, as:

$$\mathbf{B}_{\mathrm{bit}}^{n} = \left(\mathbf{Y}_{\mathrm{bit}}^{n} \circ \mathbf{Y}_{\mathrm{bit}}^{n}\right) * \mathbf{G} + 2\left(\mathbf{Y}_{\mathrm{sensed}}^{n+1} \circ \mathbf{Y}_{\mathrm{bit}}^{n}\right) * \mathbf{G} \tag{16}$$

and for $\mathbf{C}_{\mathrm{sensed}}^{n}$:

$$\mathbf{C}_{\mathrm{bit}}^{n} = \left(\mathbf{X}_{\mathrm{bit}}^{n} \circ \mathbf{Y}_{\mathrm{bit}}^{n}\right) * \mathbf{G} + \left(\mathbf{X}_{\mathrm{sensed}}^{n+1} \circ \mathbf{Y}_{\mathrm{bit}}^{n}\right) * \mathbf{G} + \left(\mathbf{Y}_{\mathrm{sensed}}^{n+1} \circ \mathbf{X}_{\mathrm{bit}}^{n}\right) * \mathbf{G}\,. \tag{17}$$

The following step of the algorithm involves the computation of the increment of (7) and (8) which, similarly as before, can be derived as:

$$[\mathrm{Tr}(\mathbf{M})]_{\mathrm{bit}}^{n} = \mathbf{A}_{\mathrm{bit}}^{n} + \mathbf{B}_{\mathrm{bit}}^{n} \tag{18}$$

$$[\det(\mathbf{M})]_{\mathrm{bit}}^{n} = \left(\mathbf{A}_{\mathrm{bit}}^{n} \circ \mathbf{B}_{\mathrm{bit}}^{n}\right) + \left(\mathbf{A}_{\mathrm{sensed}}^{n+1} \circ \mathbf{B}_{\mathrm{bit}}^{n}\right) + \left(\mathbf{B}_{\mathrm{sensed}}^{n+1} \circ \mathbf{A}_{\mathrm{bit}}^{n}\right) - \left[\left(\mathbf{C}_{\mathrm{bit}}^{n} \circ \mathbf{C}_{\mathrm{bit}}^{n}\right) + 2\left(\mathbf{C}_{\mathrm{sensed}}^{n+1} \circ \mathbf{C}_{\mathrm{bit}}^{n}\right)\right]. \tag{19}$$

Finally, similarly as before, the derivation of $\mathbf{R}_{\mathrm{bit}}^{n}$ [increment for (9)] is:

$$\mathbf{R}_{\mathrm{bit}}^{n} = [\det(\mathbf{M})]_{\mathrm{bit}}^{n} - k\left[\left([\mathrm{Tr}(\mathbf{M})]_{\mathrm{bit}}^{n} \circ [\mathrm{Tr}(\mathbf{M})]_{\mathrm{bit}}^{n}\right) + 2\left([\mathrm{Tr}(\mathbf{M})]_{\mathrm{sensed}}^{n+1} \circ [\mathrm{Tr}(\mathbf{M})]_{\mathrm{bit}}^{n}\right)\right]. \tag{20}$$

Notice that (15)-(20) provide an incremental computation framework for the results needed in order to derive the increment $\mathbf{R}_{\mathrm{bit}}^{n}$ of the output salient point matrix $\mathbf{R}_{\mathrm{sensed}}^{n}$. The remaining step in order to complete the derivation of the salient points for the input image $\mathbf{I}_{\mathrm{sensed}}^{n}$ (i.e. the results from the image sensed up to, and including, bitplane n) is the addition of the increment of the results to their previous counterparts, i.e.:

$$\mathbf{Q}_{\mathrm{sensed}}^{n} = \mathbf{Q}_{\mathrm{sensed}}^{n+1} + \mathbf{Q}_{\mathrm{bit}}^{n} \tag{21}$$

with $\mathbf{Q} \in \{\mathbf{X}, \mathbf{Y}, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathrm{Tr}(\mathbf{M}), \det(\mathbf{M}), \mathbf{R}\}$. This process can then proceed to the final salient point selection in order to derive the salient points when the image is sensed up to (and including) bitplane n. This selection occurs following the technique used in the original algorithm, i.e. as described in Subsection II.B.

The process can be continued in the same manner for each subsequent bitplane $n-1, n-2, \ldots, 0$ captured by the image sensor. If the computation or energy resources provided by the realization platform are exhausted, or when the derived salient points are considered sufficient for the particular application, the salient point detection can be terminated. For each bitplane n the incremental approach derives the same results as the conventional approach that processes all $N-n$ bitplanes simultaneously.

The derivation of (15)-(20) involves several calculations that are repeated for each input bitplane n. However, the dynamic range of the inputs is considerably smaller than the inputs of the conventional approach that consist of samples with $N-n$ bits (for terminating bitplane n). Still, this process can require considerable computational overhead compared to the original algorithm.

By observing the indicative results of Figure 1, one notices that the derived salient points for each bitplane tend to be clustered around the areas of previously-derived salient points from the higher bitplanes. This is to be expected as strong salient points are likely to be detected early on during the sensing process and then remain localized within a certain region as additional bitplanes are added to the input image. For this reason, we introduce an adaptive image scanning mask which focuses the image sensing and the processing around the areas of previously-derived salient points. Specifically, once the set:

$$\mathfrak{s}^n = \left\{ s_1^n \quad s_2^n \quad \cdots \quad s_{T(n)}^n \right\} = \left\{ (\mathrm{r}_1^n, \mathrm{c}_1^n) \quad (\mathrm{r}_2^n, \mathrm{c}_2^n) \quad \cdots \quad (\mathrm{r}_{T(n)}^n, \mathrm{c}_{T(n)}^n) \right\} \tag{22}$$

of $T(n)$ total salient points is derived for bitplane n (with each element $(\mathrm{r}_t^n, \mathrm{c}_t^n)$, $1 \leq t \leq T(n)$, representing the coordinates of salient point s_t^n), we create a binary mask $\mathbf{W}_{n-1}$ with elements initially set to “false,” i.e. for each element $w_{n-1}[r,c]$ of $\mathbf{W}_{n-1}$: $w_{n-1}[r,c] = 0$. Then we set to “true” all the elements of $\mathbf{W}_{n-1}$ in the window of $Z \times Z$ pixels around each salient point s_t^n, $1 \leq t \leq T(n)$, of bitplane n. This is done using the following rule:

$$\forall w_{n-1}[r,c] \in \mathbf{W}_{n-1} : \begin{cases} w_{n-1}[r,c] = 1, \mathit{iff} \ \left(\exists k : |r - \mathrm{r}_k^n| \leq Z/2 \ \ \mathit{and} \ \ |c - \mathrm{c}_k^n| \leq Z/2 \right) \\ w_{n-1}[r,c] = 0, \mathrm{otherwise} \end{cases}. \tag{23}$$

The selection of the window-size parameter Z will be discussed in the experimental section of the paper. Prior to sensing or processing any bit $i_{\mathrm{bit}}^{n-1}[r,c]$ contained in the input bitplane $\mathbf{I}_{\mathrm{bit}}^{n-1}$ with (11) or (12), we test whether $w_{n-1}[r,c] = 1$, i.e. whether the point lays in an area around a previously-found salient point. The steps to perform the sensing and computation with the proposed approach are summarized in the pseudocode of Table 1.

```
For each bitplane $n$, $n = N-1, N-2, \ldots, 0$
  For each point $(r,c)$, $0 \le r < R$, $0 \le c < C$
    If $n = N-1$        // highest bitplane: sense all bits and the sign bit
      Sense $i^N_{\text{bit}}[r,c]$ (sign bit) and $i^{N-1}_{\text{bit}}[r,c]$
    Else if $w_n[r,c] = 1$  // sensing occurs only within the scanning mask
      Sense $i^n_{\text{bit}}[r,c]$
    Else
      Set $i^n_{\text{bit}}[r,c] = 0$
  Calculate $\mathbf{X}^n_{\text{bit}}, \mathbf{Y}^n_{\text{bit}}, \mathbf{A}^n_{\text{bit}}, \mathbf{B}^n_{\text{bit}}, \mathbf{C}^n_{\text{bit}}, [\mathrm{Tr}(\mathbf{M})]^n_{\text{bit}}, [\det(\mathbf{M})]^n_{\text{bit}}, \mathbf{R}^n_{\text{bit}}$ using eq. (11)-(20)
  If $n < N-1$    // if not at the maximum bitplane, create the sensed results
    Apply the update process of eq. (21)
  Derive the salient point set $\mathfrak{s}^n$ corresponding to image $\mathbf{I}^n_{\text{sensed}}$ (Subsection II.B)
  If $n > 0$       // create the scanning mask to be used for the next bitplane
    For each point $(r,c)$, $0 \le r < R$, $0 \le c < C$
      Apply the rule of eq. (22) to derive $w_{n-1}[r,c]$
```

Table 1. Pseudocode for the proposed adaptive sensing and processing for the salient point detection for each input bitplane n.

III. COMPUTATIONAL AND SENSING CHARACTERISTICS

In this section we first present the metrics used for quantification of the required computation and sensing energy for each configuration of the SPD (Subsection III.A and III.B). We then propose stochastic source models encapsulating the statistics of the input and output of the SPD (Subsection III.C). These models are used in order to derive the final estimates for the expected computation and energy requirements for incremental salient point detection in images (Subsection III.D).

A. Implementation Complexity of Variable Bit-width Arithmetic Operations

We are quantifying the differences of the conventional computation of the salient-point detection algorithm of Subsection II.A and II.B versus the proposed approach of Subsection II.C. These differences will be studied in terms of the computational effort required to complete the detection task, whether it is for a single bitplane or for the entire set of bitplanes. In this respect, a metric used commonly in the literature is the required number of additions and multiplications. However, arithmetic operations in the proposed incremental refinement approaches deal with data with significantly-reduced bitwidth in comparison to the conventional computation of the salient-point detection algorithm that processes all bitplanes at once. To distinguish this effect, we follow the approach proposed in our previous work [14] and utilize the following metrics inspired by classic research work in the area-time complexity of binary multiplication and addition [15] [16].

Definition 3 (from [14]): Addition of two numbers represented with N_1 and N_2 bits, each having an additional bit as the sign bit as in (10), requires the following number of operations:

$$\mathrm{Cost}_{\text{add}} = \begin{cases} \max\{N_1, N_2\} + 1 & \text{, if both numbers are nonzero} \\ 0 & \text{, otherwise} \end{cases} \tag{24}$$

□

Definition 4 (from [14]): Multiplication of two numbers represented with N_1 and N_2 bits, each having an additional bit as the sign bit as in (10), requires the following number of operations:

$$\text{Cost}_{\text{mult}} = \begin{cases} (\max\{N_1, N_2\} + 1) \cdot (\min\{N_1, N_2\})^{1+\xi} & \text{, if both numbers are nonzero} \\ 0 & \text{, otherwise} \end{cases} \qquad (25)$$

with $\xi \geq 0$ a system parameter indicating how "hard" is binary multiplication in comparison to binary addition. □

As discussed in our previous work [14], they can be intuitively viewed as follows: Assume a virtual processing element (PE) able to perform signed addition between two bits and the carry information. Starting from the LSB, addition is (maximally) requiring $\max\{N_1, N_2\} + 1$ activations of the PE for two numbers with N_1 and N_2 bits. Similarly, by viewing multiplication as cumulative additions, the number of activations of the PE is given by (25), with the system parameter ξ indicating the cost of accumulating the intermediate results. If any of the two operands is zero, no operations are required (apart from a minimal "zero detection" effort), since the result is trivial.

B. Energy Requirements for Adaptive Image Sensing following the Binary Mask $\mathbf{W}_n$

If we assume that sensing an individual bit using successive analog-to-digital sensors (e.g. the CMOS-based image sensor of Yang et al [13]) consumes E_{bit} Joules, we can derive the overall energy consumption of the conventional approach for the computation of the detector up to (and including) bitplane n as:

$$\text{Energy}_{\text{conventional}}(n) = R \cdot C \cdot (N - 1 - n) \cdot E_{\text{bit}} \text{ Joules.} \qquad (26)$$

The last equation shows a linear increase with the number of bitplanes. Notice that the assumption of constant E_{bit} means that the information requirements (number of image bits retrieved from the sensor) are linearly related to the energy consumption of the sensing process.

On the other hand, the proposed approach involves adaptive sensing for each individual bitplane n based on the binary mask $\mathbf{W}_n$. The sensing requirements are:

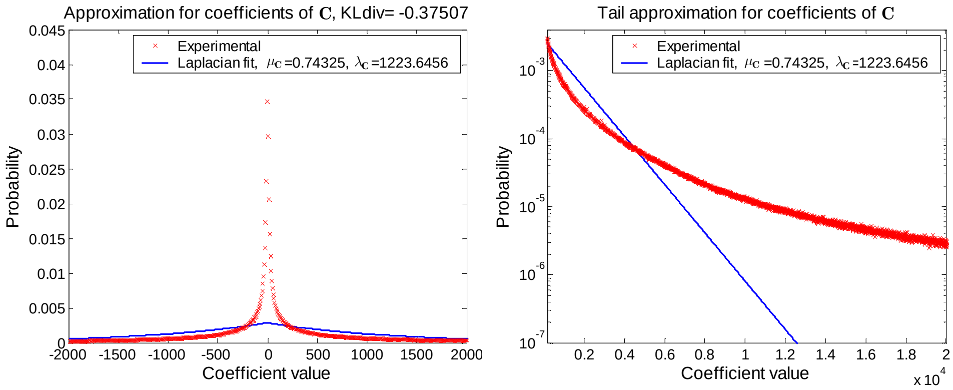
$$\text{Energy}_{\text{incremental}}(n) = \left(R \cdot C + \sum_{i=n}^{N-2} \sum_{r=1}^{R} \sum_{c=1}^{C} w_i[r,c] \right) \cdot E_{\text{bit}} \text{ Joules} \qquad (27)$$

since we sense all bits of bitplane $N-1$ and then sense bitplanes $N-2,\ldots,n$ according to binary masks $\mathbf{W}_{N-2},\ldots,\mathbf{W}_n$. If the binary mask $\mathbf{W}_i$ does not cover the entire input bitplane of the image for each $n \leq i \leq N-2$, we expect the energy requirements of the proposed approach to be less than the energy required by the conventional approach. We explore the energy performance of both approaches in the experimental section of the paper.

C. Stochastic Source Models for Matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, *and Output Salient Matrix* $\mathbf{R}$

Having defined the computational and energy metrics used in this paper, in this subsection we complete the description by considering the source statistics. These are important in the proposed framework because the input source "drives" the system operation by appropriately activating the sensor and the computation engine. Hence, based on appropriate source models, we can estimate the computational and energy requirements for the proposed framework as well as for the conventional realization of the SPD that senses the image up to a certain bitplane and then derives the salient points corresponding to this accuracy.

Starting with the distribution of the coefficients a, b, c of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ (respectively), there is a variety of stochastic image models in the related literature that could be investigated [17]. Most models analyzing filtered input images with derivative-based filters assume some form of joint distribution of pixels [17], or a doubly-stochastic model [18], in order to represent the correlation of neighboring coefficients. Lam and Goodman [18] assume that transform coefficients are independently Gaussian distributed conditional to their variances. Assuming an exponential distribution of the coefficient variances within a localized area of 8×8 coefficients, the resulting marginal distribution turns out to be Laplacian. Figure 2 shows the fitted Laplacian distribution to the experimental probability mass function (PMF) of the coefficients of $\mathbf{C}$. This distribution fails to model the sharp peak of the PMF. At the same time, the tail of the distribution decays too quickly in comparison to the PMF data. Similar results were obtained for the coefficients of matrices $\mathbf{A}$ and $\mathbf{B}$, with the difference that the Laplacian distribution becomes an exponential distribution since the coefficients of these matrices are non negative. All 215 images of the USC test-set [19] were used for the derivation of the PMFs corresponding to the coefficients $h \in \{a,b,c\}$ of $\mathbf{H} \in \{\mathbf{A},\mathbf{B},\mathbf{C}\}$.


Approximation for coefficients of $\mathbf{C}$, KLdiv= -0.37507

Tail approximation for coefficients of $\mathbf{C}$

Figure 2. Left: experimental probability mass function and Laplacian distribution for the coefficients of $\mathbf{C}$. KLdiv is the obtained Kullback-Leibler divergence [20]. Right: Log plot of the tail approximation indicating the failure of the approximation.

In order to capture the statistics of the coefficients of $\mathbf{H} \in \{\mathbf{A},\mathbf{B},\mathbf{C}\}$ more accurately, in this paper we extend the Laplacian model by using a mixture of $L_{\mathbf{H}}$ distributions:

$$H \sim \mathrm{P}(h) = \sum_{i=1}^{L_{\mathbf{H}}} \frac{w_{\mathbf{H}i}}{f_{\mathbf{H}}\lambda_{\mathbf{H}i}} \exp\left\{-\frac{|h-\mu_{\mathbf{H}i}|}{\lambda_{\mathbf{H}i}}\right\} \tag{28}$$

with:

$$f_{\mathbf{H}} = \begin{cases} 1, \text{ for } \mathbf{H} \in \{\mathbf{A},\mathbf{B}\} \\ 2, \text{ for } \mathbf{H} = \mathbf{C} \end{cases} \quad \text{and } \forall i : \mu_{\mathbf{H}i} \equiv 0\,, \tag{29}$$

where $\lambda_{\mathbf{H}i}$ and $\mu_{\mathbf{H}i}$ are the parameters of the ith component of the mixture. Finally, the weights $w_{\mathbf{H}i}$ are forming the final mixture of the $L_{\mathbf{H}}$ distributions. The factors set in (29) ensure that $\mathrm{P}(a)$, $\mathrm{P}(b)$ become a mixture of exponential distributions. Instead of deriving the distributions of (28) simply by curve fitting, we used the expectation maximization (EM) algorithm to obtain the maximum likelihood estimate for the parameters of (28). The results are presented in Figure 3, while the corresponding parameters are given in Table 2. For the generation of the PMFs using (2)-(4), a Gaussian window of 12×12 pixels was used [derived by (5) with $\sigma^2 = 2$ (recommended parameter for the Harris detector [2])]. The derived mixture model fits the data much more accurately than the single Laplacian distribution. This is also demonstrated for the coefficients of $\mathbf{C}$ by the significantly-reduced Kullback-Leibler divergence reported in Figure 3 in comparison to the one reported in Figure 2.

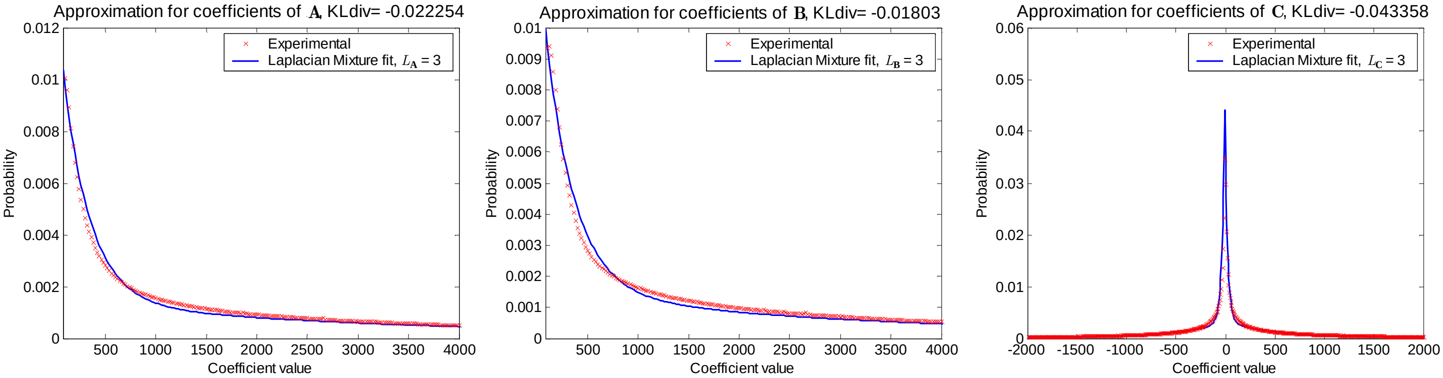


Figure 3. Experimental probability mass functions and Laplacian mixture fit for the coefficients of matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$. KLdiv is the obtained Kullback-Leibler divergence for each case [20].

i	$\mu_{\mathbf{A}i}$	$\lambda_{\mathbf{A}i}$	$w_{\mathbf{A}i}$	$\mu_{\mathbf{B}i}$	$\lambda_{\mathbf{B}i}$	$w_{\mathbf{B}i}$	$\mu_{\mathbf{C}i}$	$\lambda_{\mathbf{C}i}$	$w_{\mathbf{C}i}$
1	0.000	16422.636	0.305	0.000	13411.619	0.315	0.000	3316.214	0.317
2	0.000	2801.561	0.368	0.000	2634.697	0.361	0.000	432.324	0.366
3	0.000	267.208	0.327	0.000	287.747	0.324	0.000	26.993	0.317

Table 2. Parameters for the distribution of coefficients in $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ corresponding to $L_{\mathbf{A}} = L_{\mathbf{B}} = L_{\mathbf{C}} = 3$.

By expanding (9) we can derive the final salient point matrix $\mathbf{R}$ parametrical to matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ as:

$$\mathbf{R}^n_{\text{sensed}} = (1-2k)\mathbf{A}^n_{\text{sensed}} \circ \mathbf{B}^n_{\text{sensed}} - \mathbf{C}^n_{\text{sensed}} \circ \mathbf{C}^n_{\text{sensed}} - k(\mathbf{A}^n_{\text{sensed}} \circ \mathbf{A}^n_{\text{sensed}} + \mathbf{B}^n_{\text{sensed}} \circ \mathbf{B}^n_{\text{sensed}}) . \qquad (30)$$

The last equation indicates that the distribution of the random variables R corresponding to the final coefficients r of $\mathbf{R}$ is derived based on the individual distributions of the coefficients in $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$. However, since the individual probability distributions are co-dependent and the combination of their random variables corresponding to (30) will lead to a very complicated expression, we opt to experimentally fit the probability mass function of the resulting coefficients of $\mathbf{R}$ derived by a variety of natural images [19]. Following the same model as in (28) and by using the EM algorithm we derived the result seen in Figure 4. The corresponding parameters are presented in Table 3. The use of the mixture model of $L_{\mathbf{R}} = 4$ Laplacian distribution ensures we capture the tail of the PMF accurately. This is exhibited in Figure 5 where we display the fit of the derived mixture models for the tail of the distribution of the coefficients of $\mathbf{C}$ and $\mathbf{R}$. The comparison of the fit for the tail of $\mathbf{C}$ presented in Figure 5 with the one shown in the right side of Figure 2 demonstrates the superiority of the mixture model in comparison to the conventional single Laplacian distribution.

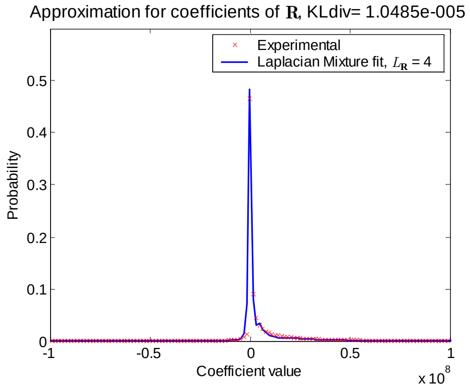


Figure 4. Experimental probability mass function and Laplacian mixture fit for the coefficients of matrix $\mathbf{R}$. KLdiv is the obtained Kullback-Leibler divergence [20].

i	$\mu_{\mathbf{R}i}$	$\lambda_{\mathbf{R}i}$	$w_{\mathbf{R}i}$
1	-21775.500	785703.000	0.555
2	4692442.500	3761440.475	0.167
3	21977908.500	20878192.930	0.177
4	174404290.500	49684133552.171	0.101

Table 3. Parameters for the distribution of the coefficients in $\mathbf{R}$ corresponding to $L_{\mathbf{R}} = 4$.

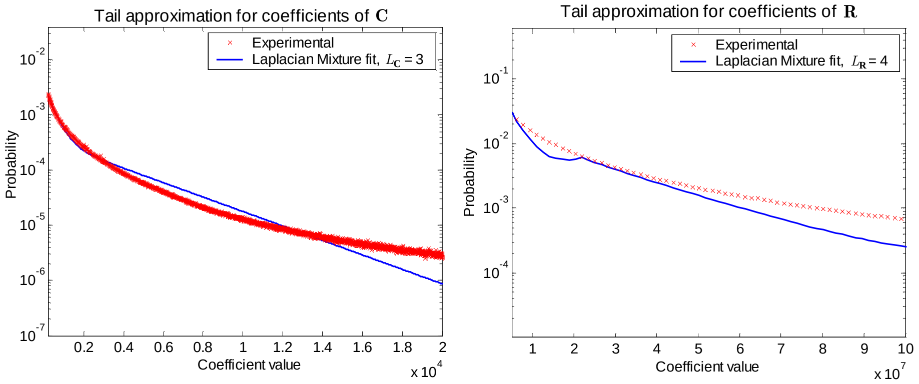


Figure 5. Log plot of the tail of the distribution of $\mathbf{C}$ and $\mathbf{R}$.

D. *Expected Sensing Energy and Computational Requirements*

We complete this section by estimating the number of required activations for both the conventional and the proposed approach using the stochastic source models proposed in the previous subsection.

Starting with the proposed approach, we produce the results incrementally following (11)-(20). For each input bitplane n, the probability of activation of the sensor and the computational units is dependent on two factors:

i) the probability of having non-zero inputs within one or more of the $\mathbf{A}^{n}_{\text{bit}}$, $\mathbf{B}^{n}_{\text{bit}}$, $\mathbf{C}^{n}_{\text{bit}}$ matrices for each sensed bitplane n;

ii) the probability of having non-zero binary mask $\mathbf{W}_n$.

Part (i) expresses the probability of activation of the computational units due to the non-zero input. Part (ii) on the other hand expresses the percentage of the sensed bitplane n.

Starting with part (i), the probability of a coefficient $h_{\text{bit}}^{n} \in \{a_{\text{bit}}^{n}, b_{\text{bit}}^{n}, c_{\text{bit}}^{n}\}$ of $\mathbf{H}_{\text{bit}}^{n}$ ($\mathbf{H} \in \{\mathbf{A},\mathbf{B},\mathbf{C}\}$) being zero is:

$$\Pr[h_{\text{bit}}^{n} = 0] = \sum_{i=1}^{L_{\mathbf{H}}} w_{\mathbf{H}i} \exp\left\{\frac{-9\left[2^{N+n+1} - 3 \cdot 2^{2n}\right]}{\lambda_{\mathbf{H}i}}\right\} \tag{31}$$

The proof of (31) is contained in Appendix I.

Concerning part (ii), we determine the probability of a coefficient r_{sensed}^{n} of $\mathbf{R}_{\text{sensed}}^{n}$ being above the detector threshold Θ or below $-\Theta$ as:

$$\Pr[|r_{\text{sensed}}^{n}| \geq \Theta] = \frac{1}{2}\sum_{i=1}^{L_{\mathbf{R}}} w_{\mathbf{R}i}\left(\exp\left\{-\frac{\Theta - \mu_{\mathbf{R}i}}{\lambda_{\mathbf{R}i}}\right\} + \exp\left\{-\frac{\Theta + \mu_{\mathbf{R}i}}{\lambda_{\mathbf{R}i}}\right\}\right) \tag{32}$$

The proof of (32) is given in Appendix II with the threshold Θ given by (41).

The derivations of (31) and (32) allow for a theoretical characterization of the required computational and energy requirements for the proposed approach. The expected computation for the proposed incremental approach up to (and including) bitplane n is:

$$\text{Cost}_{\text{incremental}}^{\text{model}}(n) = R \cdot C \cdot \sum_{i=n}^{N-1}\left[\left(1 - \prod_{h_{\text{bit}}^{i} = \{a_{\text{bit}}^{i}, b_{\text{bit}}^{i}, c_{\text{bit}}^{i}\}} \Pr[h_{\text{bit}}^{i} = 0]\right)\Pr[|r_{\text{sensed}}^{i+1}| \geq \Theta]\right] \cdot \text{Cost}_{\text{ops}} \tag{33}$$

where Cost_{ops} represents the average cost in terms of arithmetic operations for each non-zero input. In (33), for each bitplane i we combine the probability of having non-zero mask, i.e. $\Pr[|r_{\text{sensed}}^{i+1}| \geq \Theta]$ (for the previously-derived result of bitplane $i+1$), and the probability that the coefficients a_{bit}^{i}, b_{bit}^{i}, c_{bit}^{i} are not zero. These are the conditions that will activate the computational engine for each bitplane i ($n \leq i \leq N-1$), according to part (i) and part (ii) presented before. We remark that since for the MSBs of the image we are scanning the entire bitplane, we set: $\Pr[|r_{\text{sensed}}^{N}| \geq \Theta] \equiv 1$.

Similarly, concerning the expected sensing energy we have:

$$\text{Energy}_{\text{incremental}}^{\text{model}}(n) = R \cdot C \cdot \sum_{i=n}^{N-1} \Pr[|r_{\text{sensed}}^{i+1}| \geq \Theta] \cdot E_{\text{bit}} \text{ Joules} \tag{34}$$

Finally, concerning the conventional approach, the probability of activation of the computational units is dependent on the probability of a coefficient $h_{\text{sensed}}^{n} \in \{a_{\text{sensed}}^{n}, b_{\text{sensed}}^{n}, c_{\text{sensed}}^{n}\}$ of $\mathbf{H}_{\text{sensed}}^{n}$ ($\mathbf{H} \in \{\mathbf{A},\mathbf{B},\mathbf{C}\}$) being zero. This can be derived as:

$$\Pr[h_{\text{sensed}}^{n} = 0] = \sum_{i=1}^{L_{\mathbf{H}}} w_{\mathbf{H}i} \exp\left\{\frac{-9\left(2^{N} - 2^{n}\right)^{2}}{\lambda_{\mathbf{H}i}}\right\} \tag{35}$$

The proof of (35) is presented in Appendix III. If we assume that the terminating bitplane is n, the expected computational cost for the conventional approach is:

$$\text{Cost}_{\text{conventional}}^{\text{model}}(n) = R \cdot C \cdot \left(1 - \prod_{h_{\text{sensed}}^{n} = \{a_{\text{sensed}}^{n}, b_{\text{sensed}}^{n}, c_{\text{sensed}}^{n}\}} \Pr[h_{\text{sensed}}^{n} = 0]\right) \cdot \text{Cost}_{\text{ops}}' \tag{36}$$

where $\mathrm{Cost}'_{\mathrm{ops}}$ represents the average cost in terms of arithmetic operations for each non-zero input. In general we expect that $\mathrm{Cost}'_{\mathrm{ops}} > \mathrm{Cost}_{\mathrm{ops}}$ because the conventional approach has inputs with higher dynamic range than the proposed incremental approach. The derived models for the expected computation and energy are validated in the following section, where we present experimental results with the conventional and incremental SPD.

IV. EXPERIMENTS

We first present the utilized settings for our experimental validation (Subsection IV.A). In order to demonstrate the results of the intermediate detectors when sensing from the MSBs to the LSBs, Subsection IV.B presents indicative results with four selected images from our experimental test-set that are representative of a variety of content. Subsection IV.C shows computational and sensing-energy results corresponding to the presented SPD results and finally Subsection IV.D presents a new application for salient-point driven sensing of images. Supplementary results to the ones presented in this paper are available on-line [27].

A. Experimental Settings

We experimented with a variety of natural images including human faces, pictures of scenery and objects as well as surveillance pictures. The utilized test-set is presented in Figure 6. All images are grayscale images with 256 intensity levels. The experimental settings for the utilized SPD were set as suggested by Schmid et al [6] in their evaluation of the Harris detector against other approaches, i.e. $k = 0.06$, $\sigma^2 = 2$, $\Theta = 0.01 \cdot \max\{\mathbf{R}^n_{\mathrm{sensed}}\}$. The setting chosen for the detector threshold Θ for each terminating bitplane n will select the points with magnitude surpassing $\theta = 1\%$ of the maximum salient point strength for that bitplane. In general, the SPD derived with these settings is seldom optimal for any given image, but it is reported to perform well on average on collections of different images [6]. Concerning parameters specific for the proposed incremental approach, we set the size of the window parameter for the sensing mask around each salient point to: $Z = 80$ pixels for $n = 7$, $Z = 60$ pixels for $n = 6$, $Z = 50$ pixels for $n = 5$, and $Z = 30$ for $0 \le n \le 4$. These settings were chosen such that identical results between the proposed incremental approach and the conventional algorithm were obtained for all our experiments. We observed that for each bitplane n this occurs when the size of the window for this bitplane is at least eight times larger than the maximum Chamfer distance[1] between the detected salient points of bitplane n and the detected salient points of the minimum bitplane.

---

[1] measured over all the images in the test set – the explicit definition of this distance is given in the next subsection

Image 01 Image 02 Image 03 Image 04

Image 05 Image 06 Image 07 Image 08

Image 09 Image 10 Image 11 Image 12

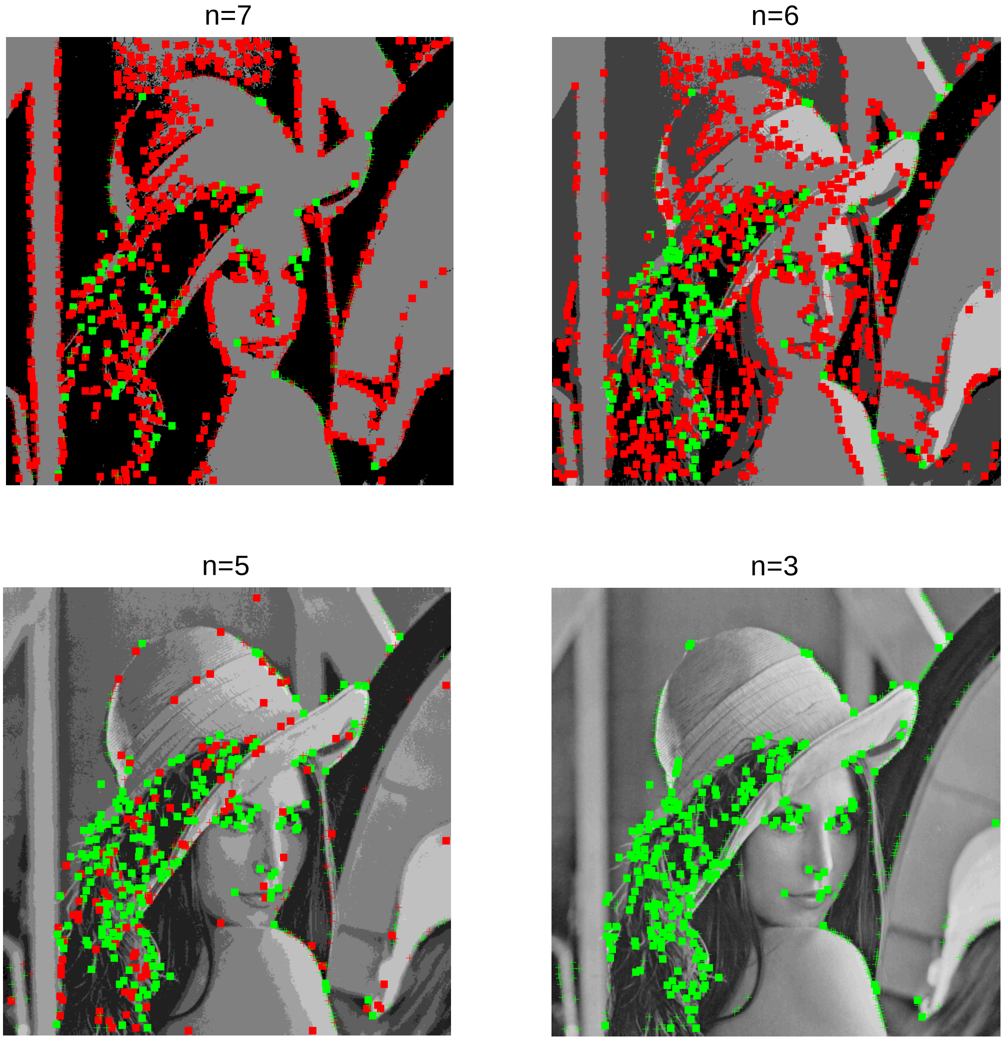
Image 13 Image 14 Image 15 Image 16

Figure 6. Sample images selected for our simulations.

With the utilized settings, the derived points stay constant for all bitplanes below $n = 3$. In some images, we observed that a small number of salient points is displaced by one pixel when sensing and processing below $n = 3$ or a small number of points (1~3) may not surpass the threshold and hence be removed. However, these minor discrepancies do not justify the additional cost of sensing and processing additional bitplanes below $n = 3$ for the SPD process. Finally, concerning the arithmetic operations in the SPD, we set the precision for fixed-point software implementation to 14 bits for the fractional part and 50 bits for the integer part (including the sign bit) for both the proposed incremental approach and the conventional algorithm. Under this accuracy, we obtained complete agreement of the derived salient points with the floating-point Matlab simulation of the Harris detector for all our experiments. We remark that not all of the bits of the integer part were used (since the dynamic range does not reach $\pm 2^{49}$). This does not affect the computation results measured based on Definition 3 and Definition 4 since the number of bits used for each fixed-point addition or multiplication of two operands is extracted based on the operands' values.

B. *Results with Incremental Salient-point Detection*

Figure 7 and Figure 8 present indicative results for bitplanes: $n = 7$, $n = 6$, $n = 5$, $n = 3$, i.e. with the intermediate detectors when subsets of the input source are used. For each bitplane, the detected salient points (corners: square marks; edges: cross marks) that correspond to the final salient points detected for the maximum source precision (when all image bitplanes are produced by the sensor) are indicated by green-colored marks. All other detected points are indicated by red-colored marks. Amongst the latter, we notice that most tend to remain in similar positions but certain points shift according to the new source information obtained from each input bitplane and may become final salient points. In some cases, groups of new salient points may appear from one bitplane to the next due to the appearance of new shades of illumination. These points usually do not correspond to final salient points. An evident example of this is given in Figure 8 ("Image 12") by comparing the result for $n = 7$ with the result for $n = 6$. This is the reason that an appropriately large window must be set for the sensing mask around each salient point [parameter Z of (23)].


n=7

n=6

n=5

n=3

Figure 7. Incremental refinement of salient point detection for "Image 01" when terminating the sensing and SPD computation at $n = 7,6,5,3$. Square marks represent corner points while cross marks represent edge points.

The quality and quantity of the obtained points depends on the image content. Overall, we observe that the number of salient points tends to decrease as edges and corners of the image are refined by sensing additional bitplanes of the image. Most points tend to gradually move towards the "final" salient points as the source precision is increased. This is visually indicated in Figure 7 and Figure 8 by the reduction of red-colored marks as the terminating bitplane is decreased. As mentioned previously, changing the threshold adaptively per image will make the SPD better suited for each particular image. However, we opted for constant settings in our experiments in order to present a unified first evaluation of the intermediate detectors for various bitplanes versus the SPD of the final terminating bitplane ($n = 3$). We summarize two important aspects of the presented results:

- Both the original SPD that processes all input bitplanes simultaneously as well as the proposed incremental SPD that processes the source incrementally obtain the same points for corners and edges for all utilized source accuracies.
- The incremental approach derives these points progressively for each new input image bitplane by refining the results of the previous bitplanes. This means that the proposed incremental algorithm can terminate at any input image accuracy and provide the best results obtained for this accuracy level. In fact, all the results presented for each image were computed by executing the incremental algorithm once and extracting the output salient points after each input bitplane is processed. To the contrary, the conventional approach needs to be executed multiple times in order to produce the same results for each bitplane.

Apart from the visual inspection of the results, in order to quantitatively test the accuracy of each intermediate SPD result we use the Chamfer distance between the set of points derived for the lowest bitplane ($n = 3$) and the set of points of each individual bitplane. We define this distance for the results of each terminating bitplane n as follows. For each salient point $s_i^3 \in \mathfrak{s}^3$ of the lowest bitplane, $1 \le i \le T(3)$, we first establish the salient point $s_{j_i^*}^n \in \mathfrak{s}^n$ with the minimum Euclidean distance:

$$j_i^* = \arg \min_{j_i = 1,\dots,T(n)} \left\| s_i^3 - s_{j_i}^n \right\|. \tag{37}$$

We remark that for each salient point we are only searching amongst the points of the corresponding type (edge or corner). Then the Chamfer distance for each terminating bitplane n is defined as:

$$\mathcal{C}(n) = \operatorname*{mean}_{i=1,\dots,T(3)} \left\| s_i^3 - s_{j_i^*}^n \right\|. \tag{38}$$

In the same manner we also define the median distance for the salient points of terminating bitplane n as:

$$\mathcal{M}(n) = \operatorname*{median}_{i=1,\dots,T(3)} \left\| s_i^3 - s_{j_i^*}^n \right\|. \tag{39}$$

The derived distances for each bitplane n for the examples of Figure 7 and Figure 8 are presented in Figure 9. The corresponding results averaged for all the images of the utilized test-set are presented in Figure 10.

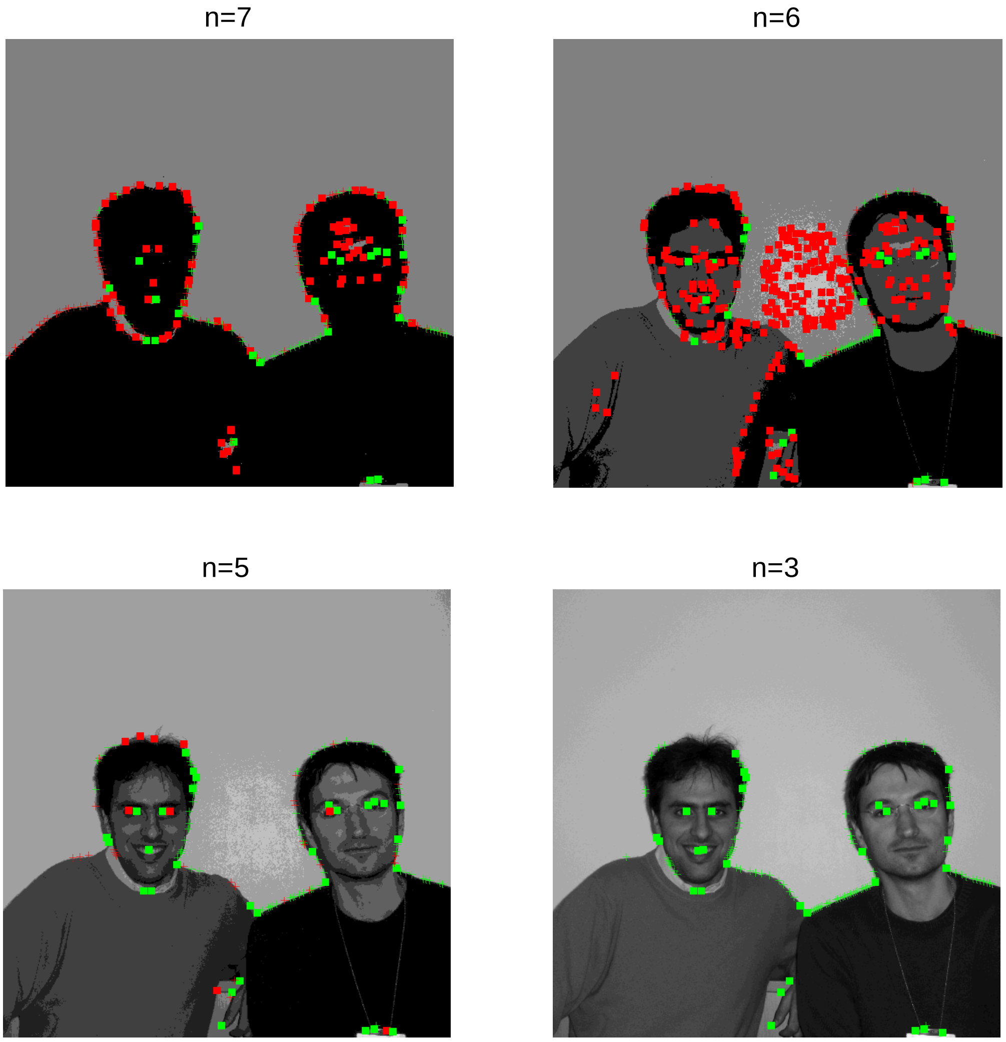


Figure 8. Incremental refinement of salient point detection for “Image 12” when terminating the sensing and SPD computation at $n = 7, 6, 5, 3$. Square marks represent corner points while cross marks represent edge points.

Overall, the distances of (38), (39) present how “different” are the selected points of each intermediate detector as compared to the final selected point. We observe that the distances between the intermediate SPDs of the various bitplanes and the final SPD of the minimum bitplane tend to decrease according to how many bitplanes are sensed and processed. There are however some notable exceptions for the Chamfer distance, as for example in the case of “Image 12” where the Chamfer distance is increasing for $n = 6$ due to the appearance of the new, temporary, corner points (see Figure 8). The median distance was found to be always non-increasing for decreasing bitplanes n. This is because the median distance is not affected by outlier distances.

Overall, we conclude that incremental refinement of salient points tends to provide results which are relevant to the final salient points selected by the detector. In addition, this relevance tends to improve with increased sensing and processing. This is a positive first step that may motivate further research in this area.

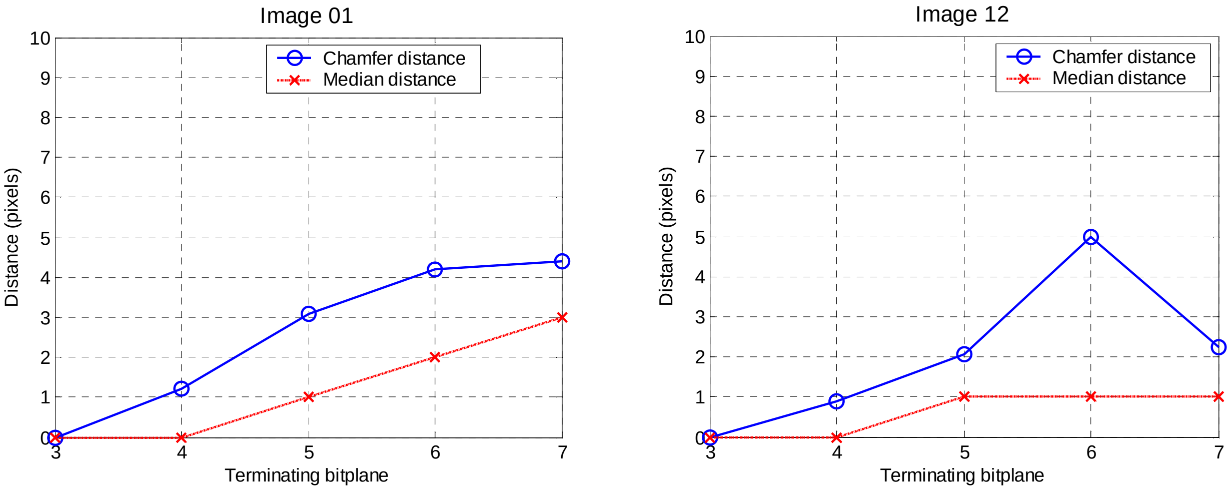


Figure 9. Chamfer and median distance (in pixels) per terminating bitplane n for the examples of Figure 7 and Figure 8.

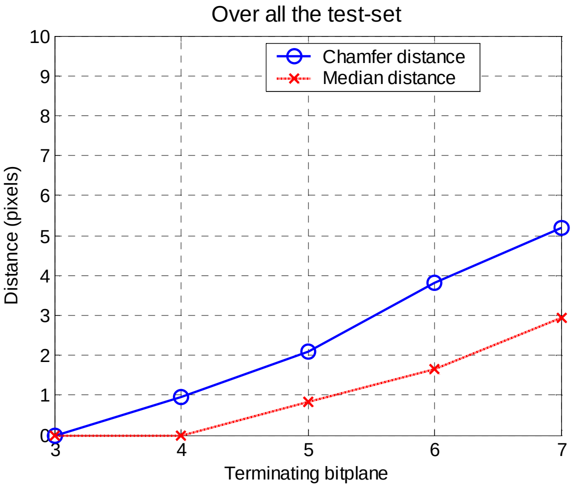


Figure 10. Chamfer and median distance (in pixels) for all the images of the test-set of Figure 6.

C. Computational and Sensing-Energy Requirements of Incremental SPD versus the Conventional SPD

We present the corresponding results for the computation and energy estimates of the proposed approach versus the conventional approach that performs the salient point detection utilizing the entire set of image bitplanes and is not refinable. This is performed by utilizing the metrics of Definition 3 and Definition 4 (where we selected $\xi = 0.5$ as a representative case) in order to measure the required computation for the proposed incremental approach and the conventional approach that processes all sensed bitplanes simultaneously and is not refinable. We also report the required sensing energy based on (26)-(27). The results are presented in Figure 11 and Figure 12. The theoretical results based on the modeling approach of Subsection III.D are presented in dashed lines for each figure. Since the proposed models utilize scaling factors (i.e. $\mathrm{Cost}_{\mathrm{ops}}$, $\mathrm{Cost}'_{\mathrm{ops}}$), they were fitted to the experimental points by linear least squares.

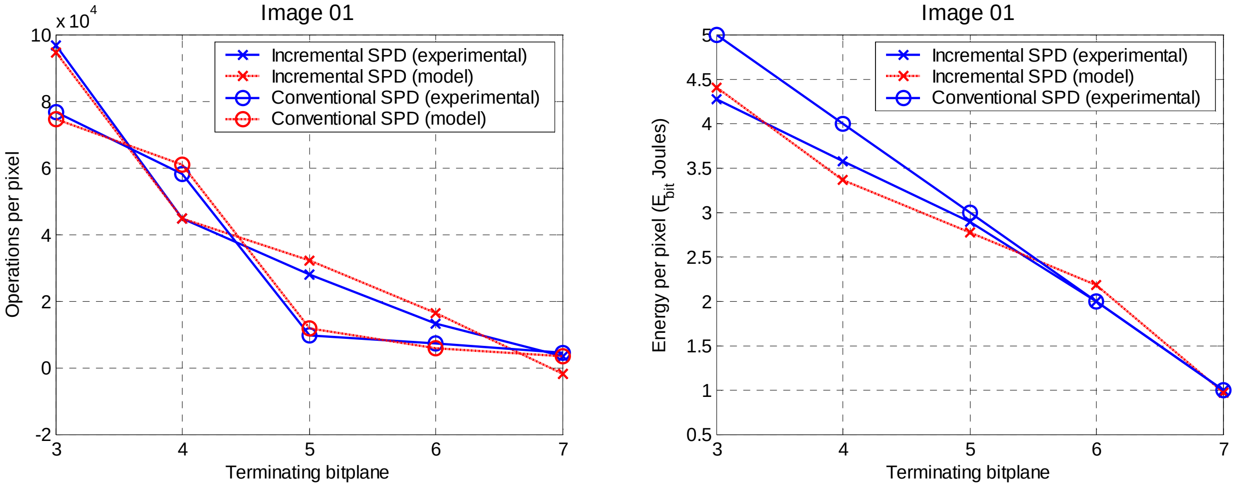


Figure 11. Computational and energy requirements corresponding to “Image 01”. Left: Operations per pixel for terminating the processing at any bitplane $n = 7,6,5,4,3$. Right: Estimated energy requirements. The dashed lines represent model estimates.

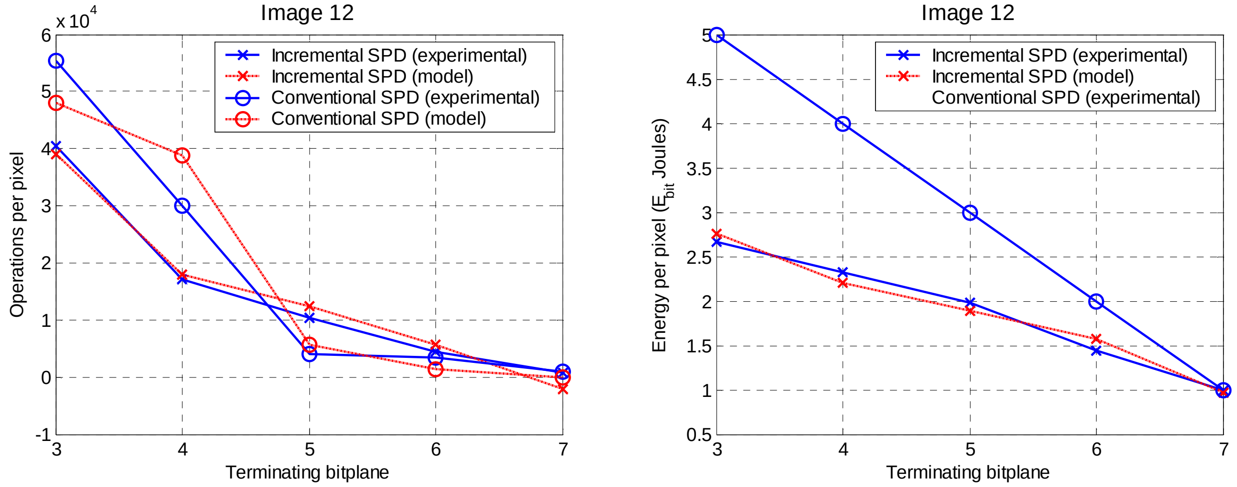


Figure 12. Computational and energy requirements corresponding to “Image 12”. Left: Operations per pixel for terminating the processing at any bitplane $n = 7,6,5,4,3$. Right: Estimated energy requirements. The dashed lines represent model estimates.

For terminating bitplanes $n = 5,6$, the results of Figure 11 and Figure 12 demonstrate that the proposed incremental approach requires increased computational resources in comparison to the conventional realization of the SPD. However, for the cases of low terminating bitplanes ($n = 3,4$), comparable computation to the conventional approach is required. This is especially true for smooth images with low texture characteristics, such as “Image 06” or “Image 12”, where the incremental SPD also appears to provide very relevant results to the final detector for $n = 3$ (see the visual results of Figure 8). In addition, we remark that the conventional approach would require significantly higher computational resources if it were to derive all the intermediate results that the incremental approach derives, since the conventional detector has to process the entire source for each newly-obtained bitplane. Concerning the sensing energy, the experimental results demonstrate that the proposed approach offers significant reductions for images with low texture characteristics (in the order of $20 \sim 50\%$), especially for the low terminating bitplanes. Only small reductions in the sensing energy are observed for images that contain textures or a large number of small scene features, such as “Image 08” (in the order of $5 \sim 10\%$).

D. A New Application: Adaptive Image Sensing based on Salient Points

Areas of an image around points of saliency are important for a variety of high-level object detection algorithms as well as for scene analysis applications. In recent years, sparse descriptors localized around salient points (and utilizing image information only around them) have been used for image retrieval [21], stereoscopic analysis [22], object detection and recognition [5] [23] and event recognition in dynamic scenes [24]. Significant work has also been conducted in the area of perceptual coding [25] as human viewers also tend to focus on areas around salient points (corners, edges). Motivated by this observation, we present an interesting application of the proposed framework by using the incremental salient-point detection process as the guide for adaptive low-energy image sensing. This is performed by changing the threshold of the SPD (from 1% up to 50% of the maximum salient-point strength) and deriving the salient points of each bitplane n followed by the binary mask $\mathbf{W}_{n-1}$ which restricts the sensing of bitplane $n-1$. In this way, we focus the sensor to the incrementally-detected points of saliency. We present indicative results in Figure 13 and Figure 14 based on the proposed approach that uses the algorithm of Table 1. The derived sensing energy percentage reported in the figures is calculated by the ratio $\mathrm{Energy}_{\mathrm{incremental}}(0) / \mathrm{Energy}_{\mathrm{conventional}}(0)$ as derived by (26), (27) for terminating bitplane $n=0$ (i.e. until the entire image is scanned by the sensor).

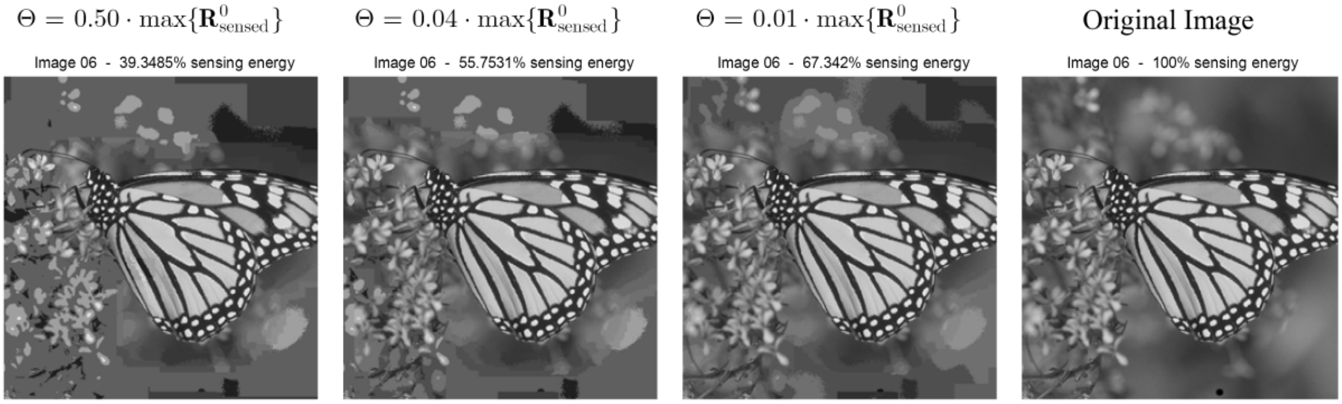

$\Theta = 0.50 \cdot \max\{\mathbf{R}^0_{\mathrm{sensed}}\}$
$\Theta = 0.04 \cdot \max\{\mathbf{R}^0_{\mathrm{sensed}}\}$
$\Theta = 0.01 \cdot \max\{\mathbf{R}^0_{\mathrm{sensed}}\}$
Original Image
Image 06 - 39.3485% sensing energy
Image 06 - 55.7531% sensing energy
Image 06 - 67.342% sensing energy
Image 06 - 100% sensing energy

Figure 13. Adaptive image sensing results for “Image 06”.

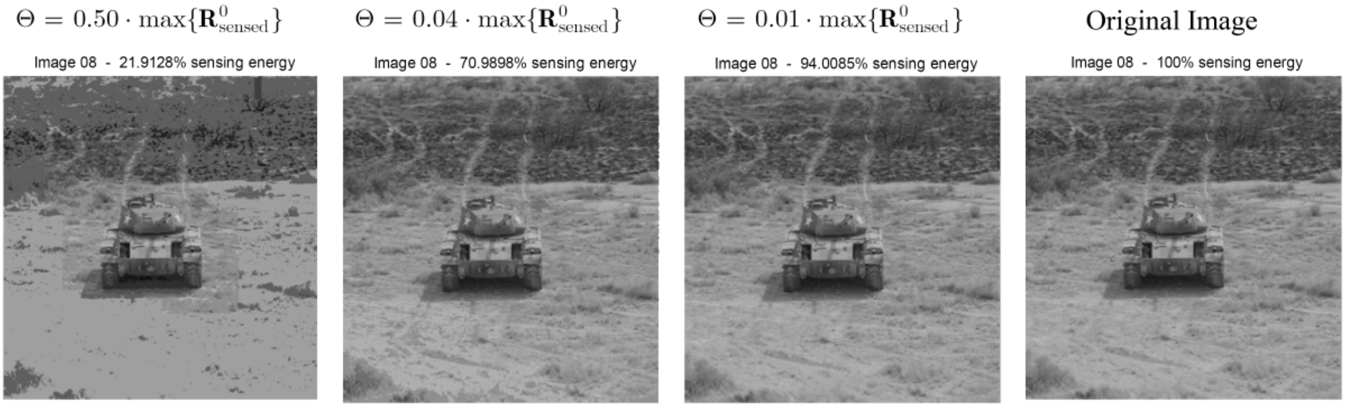

$\Theta = 0.50 \cdot \max\{\mathbf{R}^0_{\mathrm{sensed}}\}$
$\Theta = 0.04 \cdot \max\{\mathbf{R}^0_{\mathrm{sensed}}\}$
$\Theta = 0.01 \cdot \max\{\mathbf{R}^0_{\mathrm{sensed}}\}$
Original Image
Image 08 - 21.9128% sensing energy
Image 08 - 70.9898% sensing energy
Image 08 - 94.0085% sensing energy
Image 08 - 100% sensing energy

Figure 14. Adaptive image sensing results for “Image 08”.

By increasing the detector threshold, the sensing energy is decreased significantly. At the same time, the sensor tends to obtain only the image areas that have significant points of saliency, which typically correspond to areas of human faces, or texture and objects. Overall, many improvements could be envisaged for this scheme, for example experimenting with different window sizes for the binary mask $\mathbf{W}_{n-1}$ or adaptively changing the detector threshold (percentage of maximum point strength) and the parameter Z of the binary mask for each bitplane.

V. CONCLUSIONS

Salient points represent areas of “interest” for natural images. We investigated an approach for deriving salient points incrementally by increasing the number of bits sensed from the input image. The inherent advantage of the proposed approach is that intermediate salient-point detectors can be derived by increasing the source precision. We have analyzed the results of these detectors experimentally for a variety of images and presented initial evidence that incremental salient-point detection can successively refine the quality of the derived output. In order to quantify the computational and sensing-energy characteristics of the proposed approach versus the conventional salient point detector that processes all the input bits simultaneously and is not refinable, we proposed an approach that estimates the expected computation and sensing energy based on stochastic source models. This provides some initial analytical understanding behind the computational and energy requirements of incremental salient point detection. Finally, a novel application of adaptive image sensing based on salient points (which are derived incrementally) is proposed. This reduces the required sensing energy by adaptively activating the sensor only on the areas that higher-level scene analysis systems or human viewers find important.

Future investigations could explore the properties of other salient point detectors proposed in the literature when modified to operate in incremental mode. An interesting case would be the Lucas-Kanade detector used for feature tracking in image sequences, as proposed by Tomasi and Kanade [26]. A link with a higher-level vision system that could also operate incrementally is of high importance. Extended studies of the behavior of the detector under different parameter sets could help in creating further understanding of the properties of incremental refinement of salient point detection in various images. Moreover, we are currently investigating how information-theoretic criteria, such as the mutual information between sets of salient points detected for successive bitplanes, could be combined with the provided Chamfer distance to better characterize how the extra computation and sensing required when going from one bitplane to the next increases the detector performance. Finally, improvements in the adaptive sensing based on salient points could be envisaged in order to fine-tune such a system to human viewers or automatic scene-analysis systems.

APPENDIX I: PROOF OF (31)

The dynamic range $\{0, 2^n\}$ of $\mathbf{I}_{\text{bit}}^n$ is enlarged to $\left[0, 9\left(2^{N+n+1} - 3 \cdot 2^{2n}\right)\right]$ for the coefficients of $\mathbf{A}_{\text{bit}}^n$ and $\mathbf{B}_{\text{bit}}^n$ due to the convolution of $\mathbf{I}_{\text{bit}}^n$ with $\mathbf{D}$ in (11), (12) and the addition of the Hadamard products in (15)-(16). Similarly, the dynamic range of the coefficients of $\mathbf{C}_{\text{bit}}^n$ is $\left[-9\left(2^{N+n+1} - 3 \cdot 2^{2n}\right), 9\left(2^{N+n+1} - 3 \cdot 2^{2n}\right)\right]$. The probability of a coefficient $h_{\text{bit}}^n \in \{a_{\text{bit}}^n, b_{\text{bit}}^n, c_{\text{bit}}^n\}$ of $\mathbf{H}_{\text{bit}}^n$ ($\mathbf{H} \in \{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$) being zero is then:

$$\Pr[h_{\text{bit}}^n = 0] = 1 - \int_{9(1-f_{\mathbf{H}})\left[2^{N+n+1} - 3 \cdot 2^{2n}\right]}^{9\left(2^{N+n+1} - 3 \cdot 2^{2n}\right)} \sum_{i=1}^{L_{\mathbf{H}}} \frac{w_{\mathbf{H}i}}{f_{\mathbf{H}} \lambda_{\mathbf{H}i}} \exp\left\{-\frac{|h_{\text{bit}}^n|}{\lambda_{\mathbf{H}i}}\right\} dh_{\text{bit}}^n \tag{40}$$

where we have replaced $\forall i : \mu_{\mathbf{H}i} = 0$ based on the values of Table 2. We obtain (31) by solving the integral of (40). ■

APPENDIX II: PROOF OF (32)

Since the threshold of the detector is set according to the maximum value of $\mathbf{R}_{\text{sensed}}^n$ for each bitplane (see Subsection II.B), we need to establish the dynamic range of the coefficients of matrix $\mathbf{R}_{\text{sensed}}^n$. This can be done by tracking the increase in the dynamic range for each of the operations performed in (30). The dynamic range $\left\{0, \left[2^n, 2^N - 2^n\right]\right\}$ of $\mathbf{I}_{\text{sensed}}^n$ is enlarged to $\left[0, 9\left(2^N - 2^n\right)^2\right]$ for the coefficients of $\mathbf{A}_{\text{sensed}}^n$ and $\mathbf{B}_{\text{sensed}}^n$ and to $\left[-9\left(2^N - 2^n\right)^2, 9\left(2^N - 2^n\right)^2\right]$ for the coefficients of $\mathbf{C}_{\text{sensed}}^n$. This is due to the convolution of $\mathbf{I}_{\text{sensed}}^n$ with $\mathbf{D}$ and the Hadamard product in (13). For the coefficients of each of the Hadamard products $\mathbf{H_1}_{\text{sensed}}^n \circ \mathbf{H_2}_{\text{sensed}}^n$ seen in (30) (with $\mathbf{H_1}, \mathbf{H_2} \in \{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$), the range is $\left[0, 81\left(2^N - 2^n\right)^4\right]$. The range of the coefficients corresponding to the term $-k(\mathbf{A}_{\text{sensed}}^n \circ \mathbf{A}_{\text{sensed}}^n + \mathbf{B}_{\text{sensed}}^n \circ \mathbf{B}_{\text{sensed}}^n)$ in (30) is $\left[-162k\left(2^N - 2^n\right)^4, 0\right]$. We can similarly derive the ranges for the coefficients corresponding to the terms $-\mathbf{C}_{\text{sensed}}^n \circ \mathbf{C}_{\text{sensed}}^n$ and $(1-2k)\,\mathbf{A}_{\text{sensed}}^n \circ \mathbf{B}_{\text{sensed}}^n$. The final additions to complete $\mathbf{R}_{\text{sensed}}^n$ lead to dynamic range $\left[-81\left(2^N - 2^n\right)^4, 81(1-2k)\left(2^N - 2^n\right)^4\right]$. The probability of any coefficient r_{sensed}^n of $\mathbf{R}_{\text{sensed}}^n$ being above the threshold:

$$\Theta = \left(\theta/100\right) \cdot 81(1-2k)\left(2^N - 2^n\right)^4 \tag{41}$$

(with θ the percentage set by the salient point selection algorithm of Subsection II.B) or below $-\Theta$ is:

$$\Pr[|r_{\text{sensed}}^n| \geq \Theta] = 1 - \int_{-\Theta}^{\Theta} \sum_{i=1}^{L_{\mathbf{R}}} \frac{w_{\mathbf{R}i}}{2\lambda_{\mathbf{R}i}} \exp\left\{-\frac{|r_{\text{sensed}}^n - \mu_{\mathbf{R}i}|}{\lambda_{\mathbf{R}i}}\right\} dr_{\text{sensed}}^n \tag{42}$$

We obtain (32) by solving the integral of (42). ■

APPENDIX III: PROOF OF (35)

In Appendix II we derived the dynamic range of the coefficients a_{sensed}^n and b_{sensed}^n of $\mathbf{A}_{\text{sensed}}^n$ and $\mathbf{B}_{\text{sensed}}^n$ as $\left[0, 9\left(2^N - 2^n\right)^2\right]$. The dynamic range of coefficients c_{sensed}^n of $\mathbf{C}_{\text{sensed}}^n$ was derived as $\left[-9\left(2^N - 2^n\right)^2, 9\left(2^N - 2^n\right)^2\right]$. The probability of a coefficient $h_{\text{sensed}}^n \in \{a_{\text{sensed}}^n, b_{\text{sensed}}^n, c_{\text{sensed}}^n\}$ of $\mathbf{H}_{\text{sensed}}^n$ ($\mathbf{H} \in \{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$) being zero is then:

$$\Pr[h_{\text{sensed}}^n = 0] = 1 - \int_{9(1-f_{\mathbf{H}})\left(2^N - 2^n\right)^2}^{9\left(2^N - 2^n\right)^2} \sum_{i=1}^{L_{\mathbf{H}}} \frac{w_{\mathbf{H}i}}{f_{\mathbf{H}} \lambda_{\mathbf{H}i}} \exp\left\{-\frac{|h_{\text{sensed}}^n|}{\lambda_{\mathbf{H}i}}\right\} dh_{\text{sensed}}^n \tag{43}$$

where we have replaced $\forall i : \mu_{\mathbf{H}i} = 0$ based on the values of Table 2. We obtain (35) by solving the integral of (43). ■

ACKNOWLEDGEMENT

We would like to thank the anonymous reviewers for their comments. We also appreciate the useful suggestions for future work, especially the suggestion of information-theoretic criteria for characterizing the detection vs. computation performance of incremental salient-point detectors.

REFERENCES

[1] H. Moravec, “Obstacle avoidance and navigation in the real world by a seeing robot rover,” *Tech. Report CMU-RI-TR-80-03*, Robotics Inst. Carnegie Mellon University, Sept. 1980.

[2] C. Harris and M. Stephens, “A combined corner and edge detector,” *Alvey Vision Conf.*, vol. 1, pp. 147-151, 1988.

[3] F. Heitger, L. Rosenthaler, R. von der Heydt, E. Peterhans, and O. Kuebler, “Simulation of neural contour mechanism: From simple to end-stopped cells,” *Vision Research*, vol. 32, no. 5, pp. 963–981, May 1992.

[4] F. Mokhtarian, R. Suomela, “Robust image corner detection through curvature scale space,” *IEEE Trans. on Patt. Anal. Mach. Intel.*, vol. 20, no. 12, pp. 1376-1381, Dec. 1998.

[5] D. G. Lowe, “Object recognition from local scale-invariant features,” *Proc. IEEE Internat. Conf. on Comp. Vision*, vol. 2, pp. 1150-1157, 1999.

[6] C. Schmid, R. Mohr, and C. Bauckhage, “Evaluation of interest point detectors,” *Internat. J. of Comp. Vision*, vol. 37, no. 2, pp. 151-172, Feb. 2000.

[7] C. F. Chiasserini, and E. Magli, “Energy consumption and image quality in wireless video-surveillance networks,” *Proc. IEEE Personal, Indoor, Mobile Radio Comm.*, vol. 5, pp. 2357-2361, Sept. 2002.

[8] C. Y. Chong and S. P. Kumar, “Sensor networks: Evolution, opportunities and challenges,” *Proc. of the IEEE*, vol. 1247, no. 8, pp. 1247-1256, Aug. 2003,

[9] D. Grobe Sachs, W. Yuan, C. J. Hughes, A. Harris, S. V. Adve, D. L. Jones, R. H. Kravets, and K. Nahrstedt, “GRACE: A hierarchical adaptation framework for saving energy,” *Tech. Rep. UIUC DCS-R-2004-2409*, Computer Science Dept., Univ. of Illinois, Urbana Champaign, IL, Feb. 2004.

[10] W. Yuan and K. Nahrstedt, “Energy-efficient CPU scheduling for multimedia applications,” *ACM Trans. on Computer Syst.*, vol. 24 , no 3, pp. 292 - 331, Aug. 2006.

[11] D. L. Donoho, “Compressed sensing,” *IEEE Trans. on Info. Theory*, vol. 52, no. 4, pp. 1289-1306, April 2006.

[12] M. B. Wakin, J. N. Laska, M. F. Duartem D. Baron, S. Sarvotham, D. Takhar, K. F. Kelly, and R. G. Baraniuk, “An architecture for compressive imaging,” *Proc. IEEE Internat. Conf. Image Proc.*, ICIP, 2006.

[13] D. X. D. Yang, B. Fowler, and A. El Gamal, “A Nyquist-rate pixel-level ADC for CMOS image sensors,” *IEEE J. of Solid State Circ.*, vol. 34, no. 3, pp. 348-356, March 1999.

[14] Y. Andreopoulos and M. van der Schaar "Incremental refinement of computation for the discrete wavelet transform," *IEEE Trans. on Signal Process.*, to appear.

[15] R. P. Brent and H. T. Kung, “The area-time complexity of binary multiplication,” *J. of the Assoc. for Comp. Machin.*, vol. 28, no. 3, pp. 512-534, Jul. 1981.

[16] R. Bryant, “Graph-based algorithms for Boolean function manipulation,” *IEEE Trans. on Comput.*, vol. 35, no. 8, pp. 677-691, Aug. 1986.

[17] A. Srivastava, “Stochastic models for capturing image variability,” *IEEE Signal Proc. Mag.*, vol. 19, no. 5, pp. 63-76, Sept. 2002.

[18] E. Y. Lam and J. W. Goodman, “A mathematical analysis of DCT coefficient distributions for images,” *IEEE Trans. on Image Process.*, vol. 9, no. 10, pp. 1661-1666, Oct. 2000.

[19] *The USC-SIPI Image Database*, http://sipi.usc.edu/database/

[20] S. Kullback, R. A. Leibler, “On information and sufficiency,” *Annals of Math. Statistics*, vol. 22, no. 1, pp. 79-86, Mar. 1951.

[21] Q. Tian, N. Sebe, M. S. Lew, E. Loupias, and T. S. Huang, “Image retrieval using wavelet-based salient points,” *J. of Electron. Imag.*, vol. 10, no. 4, pp. 835-849, Oct. 2001.

[22] C. J. Taylor, “Surface Reconstruction from Feature Based Stereo” *Proc. IEEE Int. Conf. on Comput. Vision* (ICCV'03), pp. 184-201, 2003.

[23] S. Se, D. G. Lowe, and J. J. Little, “Vision-based global localization and mapping for mobile robots,” *IEEE Trans. on Robotics*, vol. 21, no. 3, pp. 364-375, Mar. 2005.

[24] A. Oikonomopoulos, I. Patras and M. Pantic, “Spatiotemporal salient points for visual recognition of human actions,” *IEEE Trans. Syst., Man, and Cybern.- Part B*, vol. 36, no. 3, pp. 710-719, Jun. 2006.

[25] J. Malo, I. Epifanio, R. Navarro, and E. P. Simoncelli, “Nonlinear image representation for efficient perceptual coding,” *IEEE Trans. on Image Process.*, vol. 15, no. 1, pp. 68-80, Jan. 2006.

[26] C. Tomasi and T. Kanade, “Detection and tracking of point features,” *Tech. Report, CMU-CS-91-132*, Department of Computer Science, Carnegie-Mellon University, April 1991.

[27] *Supplementary experiments with all test-set images*: http://www.ee.ucl.ac.uk/~iandreop/03171supplementary.pdf