A Biologically Inspired Approach for Fast Image Processing

[1]Sonya Coleman, [2]Bryan Scotney, [1]Bryan Gardiner

[1]School of Computing and Intelligent Systems, University of Ulster, Magee, BT48 7JL, Northern Ireland
[2]School of Computing and Information Engineering, University of Ulster, Coleraine, BT52 1SA, Northern Ireland

ABSTRACT

As features within an image may be present at many scales, application of feature detectors at multiple scales can improve accuracy of the detected localisation and orientation. As the scale and size of a feature detector increases, so does the computational complexity of implementation across the image domain. To address this issue we present a novel integral image for hexagonal pixel based images and associated multi-scale operator implementation that significantly speeds up the feature detection process. We demonstrate that this framework enables significantly faster computation than the use of conventional spiral convolution, the use of a neighbourhood address look-up table on hexagonal images.

1. INTRODUCTION

A common requirement for image processing tasks is to achieve real-time performance. Motivated by the real-time processing capabilities of the human vision system, we consider how three characteristics of the human vision system may be combined in order to reduce computational effort when implementing low-level image processing algorithms. Firstly, we consider the way in which visual information is captured: a small region, the fovea, within the retina, contains photoreceptor cones that are arranged in a densely packed hexagonal structure. Correspondingly, we consider digital images in which the pixels are hexagonal. Secondly, within the fovea photoreceptive fields of ganglion cells do not overlap [4]. Typically, feature extraction is achieved by convolving an operator at every location in the image and hence uses convolution neighbourhoods that overlap. In contrast, we develop a framework for which the convolution neighbourhoods do not overlap. Thirdly, the human eye can be subjected to three types of movement: *tremor*, drift, and micro-saccades [7]. As a consequence of eye tremor - rhythmic oscillations of the eye - the human vision system does not process single static images, but a series of temporal images that are slightly off-set due to these involuntary eye movements. Therefore, we use a set of similarly off-set images, each of which is partially processed using non-overlapping convolution neighbourhoods.

Although the presented framework can be used for many image processing algorithms, this paper uses edge detection as its application. In previous work [3] the finite element method was used to develop a systematic and efficient design procedure for operators for use with hexagonal images that are scalable through the use of an explicit scale parameter. A disadvantage of applying large spatial operators is the correspondingly long computation time. For traditional rectangular images, one approach developed to reduce the computational overhead associated with operator convolution is the use of integral images. Integral images provide a computationally efficient way of approximating convolution filters of any size on rectangular images, as the number of operations, and hence computation time, required to evaluate any rectangular region of an integral image is independent of the size of the region [9]. Integral images are a key aspect of the SURF detector [1] and have also been used for adaptive thresholding [2] and object detection [5].

In [8], we first presented the concept of fast edge detection using eye tremor, however this used the complete standard image. In this paper we design novel integral spiral images that, when combined with a biologically-inspired approach for operator implementation [8], provide a framework for obtaining feature maps efficiently over a range of scales. In Section 2 we present our novel integral spiral image, followed by the core 7-point operator design in Section 3. In Section 4 we describe the framework for fast processing and present results in Section 5.

2. SPIRAL IMAGES

2.1 Spiral Architecture

In the spiral architecture [6] the addressing scheme for the spiral image, denoted by *S*, originates at the centre of the image (pixel index 0) and spirals out using one-dimensional indexing. Figure 1 shows the spiral addressing scheme for the central portion of an image. Pixel 0 may be considered as a layer 0 cluster. Pixel 0, together with its six immediate neighbours indexed in a clockwise direction (pixels 1,…,6) then form a layer 1 cluster centred at pixel 0. This layer 1 cluster may then be combined with its six immediately neighbouring layer 1 clusters, the centres of which are indexed as 10, 20, 30, 40, 50 and 60, to form a layer 2 cluster centred at pixel 0 (as shown in Figure 1); the remaining pixels in each of these layer 1 clusters are indexed in a clockwise direction in the same fashion as the layer 1 cluster centred at 0, (e.g., for the layer 1 cluster centred at 30, the pixel indices are 30, 31, 32, 33, 34, 35 and 36). The entire spiral addressing scheme is generated by recursive use of the clusters; for example, seven layer 2 clusters are combined to form a layer 3 cluster. Ultimately the entire hexagonal image may be considered to be a layer L cluster centred at 0 comprising 7^L pixels. An important advantage of the spiral addressing scheme is that any location in the image can be represented by a single co-ordinate value, and hence the spiral image can be stored as a vector [6]. Spatially neighbouring pixels within any 7-pixel layer 1 cluster in the image remain neighbouring pixels in the one-dimensional image storage structure. This is a very useful characteristic when performing image processing tasks on the stored image vector, and this contiguity property lies at the heart of our approach to achieve fast and efficient processing for feature extraction.

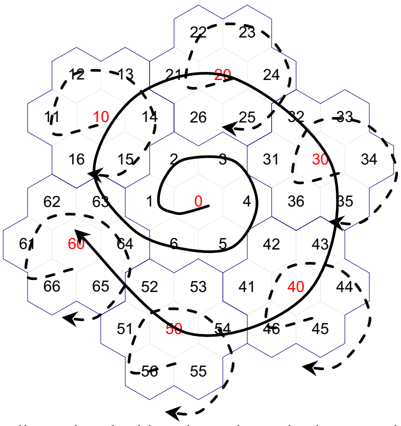


Figure 1: One-dimensional addressing scheme in the central region of the image

2.2 Integral Spiral Image

We introduce an integral spiral image, analogous to the traditional integral image approach in [9] for rectangular pixel-based images. As the spiral image S is represented by a vector, the integral spiral image, denoted by I, is computed in the following way:

$$I(p) = S(p) \quad \text{for pixel } p = 0 \tag{1}$$

$$I(p) = S(p) + I(p-1) \quad \text{for pixel } p \neq 0 \tag{2}$$

3. DESIGNING A CORE OPERATOR

The key aspect of the integral image approach is that only one 7-point operator is required (the core operator) that can be applied at multiple scales using the integral spiral image presented in Section 2. To develop the core 7-point operator we need to consider only Layer 1. We use a regular mesh of equilateral triangles with nodes placed at the pixel centres (Figure 2(a)). With each node p we associate a piecewise linear basis function ϕ_p, with $\phi_p = 1$ at node p and $\phi_m = 0$ at all other nodes $m \neq p$. Each ϕ_p is thus a "tent-shaped" function with support restricted to a small neighbourhood of six triangular elements centred at node p (Figure 2(b)). We represent the spiral image by a function $S = \sum_{q \in Q} S(q)\phi_q$, where Q denotes the set of all nodal addresses; the parameters $\{S(q)\}$ are the image intensity values at the pixel centres.

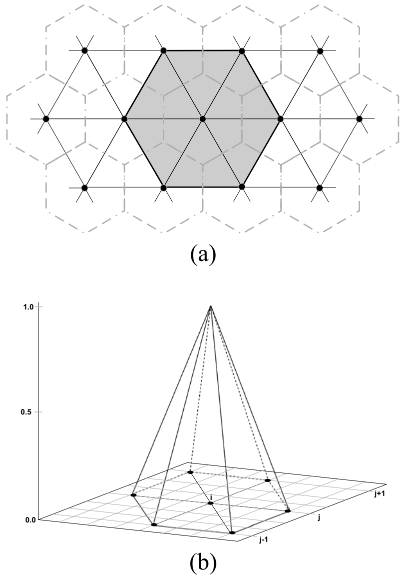


Figure 2: (a) regular mesh of equilateral triangles with nodes placed at the pixel centres; (b) "tent-shaped" function

Feature detection and enhancement operators are often based on first derivative approximations, and we consider a weak form of the first directional derivative $\partial S/\partial b \equiv \underline{b} \cdot \underline{\nabla} S$. To approximate the derivative over a layer_1 cluster centred on the pixel with spiral address p, the image derivative is multiplied by a neighbourhood test function ψ_p and the result integrated over a neighbourhood $N_1(p)$ corresponding to the layer 1 cluster centred on pixel p. Hence at pixel p we obtain a directional derivative $D_1(p)$ in any direction $\underline{b}$ ($\underline{b}$ is a unit direction vector) as

$$D_1(p) = \int_{N_1(p)} \underline{b} \cdot \underline{\nabla} S \psi_p \, d\Omega \tag{3}$$

Thus we may write

$$D_1(p) = \sum_{q \in Q} \left(S(p) \int_{N_1(p)} \underline{b} \cdot \underline{\nabla} \phi_q \psi_p^1 \, d\Omega \right) = \sum_{q \in N_1(p)} H_1(q) \times S(q) \tag{4}$$

where H_1 is the 7-point layer 1 hexagonal operator. We have chosen the neighbourhood test function ψ_p to be a Gaussian function restricted to $N_1(p)$, centred on node p and parameterised so that 95% of its central cross section falls within $N_1(p)$. The operator H_1, shown in Figure 3, is then used as the core 7-point operator.

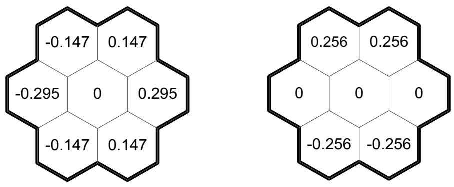


Figure 3: x- and y-components of operator H_1

4. BIOLOGICALLY-INSPIRED FRAMEWORK

4.1 Simulating Eye Tremor

Following the approach presented in [8] we consider the spiral integral image I_0 to be the "base" image, and we compute six further integral images, $I_j, j = 1,...,6$, for the same scene. The location of the origin of each of these additional images is offset spatially from I_0 by a distance of one pixel in the image plane along one of the three natural hexagonal axis directions. This mechanism simulates the phenomenon of "eye tremor". In each image $I_j, j = 1,...,6$, the pixel with spiral address "*0*" represents the same spatial location in the scene as the pixel with spiral address "*j*" in I_0. The "centre" (i.e., the pixel with spiral address zero) of each image $I_j, j = 0,...,6$, is thus located at a pixel within the layer $\lambda = 1$ neighbourhood centred at the pixel with spiral address "0" in image I_0, as shown in Figure 4.

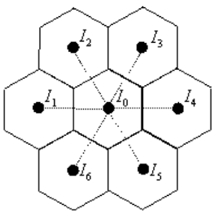


Figure 4. The 7 image centres in the eye tremor approach

Through use of the spiral architecture for pixel addressing, it is assumed that image I_0 is stored in a one-dimensional vector (with base-7 indexing). Using the spiral architecture the additional images $I_j, j = 1,...,6$, are stored similarly.

4.2 Non-Overlapping Convolution

For a given image I_0, convolution of the operator H_1 across the entire integral image plane is achieved by applying the operator sparsely to each of the seven images $I_j, j = 0,...,6$ and then combining the resultant outputs. Figure 5 shows a sample of pixels in image I_0 for which the label $j = 0,...,6$ for each pixel indicates in which of the images $I_j, j = 0,...,6$, the pixel address takes the value 0 *mod* 7. Each pixel in image I_0 may be thus uniquely labeled.

To implement the operator H_1 using an integral image, we need to determine the cluster integrals $CI(c_i)$ for the seven layer (λ-1) clusters that comprise the layer λ cluster. Here, the values of c_i denote the centres of these seven layer as $c_i = s_0 + i10^{\lambda-1}$ for $i = 0,...,6$. Using base 7 addition [7], the layer (λ-1) cluster integral value at c_i is then calculated as (λ-1) clusters. For a layer λ cluster with centre s_0, the seven corresponding layer (λ-1) cluster centres are computed

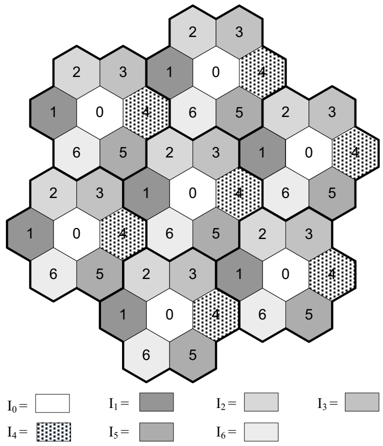


Figure 5. Pixel positions in image I_0 corresponding to pixels in images $I_j, j = 0,...,6$ with address 0 *mod* 7.

as $c_i = s_0 + i10^{\lambda-1}$ for $i = 0,...,6$. Using base 7 addition [7], the layer (λ-1) cluster integral value at c_i is then calculated as

$$CI(c_i) = I(6\sum_{k=0}^{\lambda-1} 10^k) \quad \text{for } c_i = 0 \tag{5}$$

$$CI(c_i) = I(c_i + 6\sum_{k=0}^{\lambda-1} 10^k) - I(c_i - 1) \quad \text{for } c_i \neq 0 \tag{6}$$

The operator at scale λ, applied to a layer λ cluster, is implemented by convolving a core 7-point operator with the cluster integral values $CI(c_i)$ for the seven corresponding layer (λ-1) clusters with centres c_i, $i = 0,...,6$ such that

$$D_\lambda(s) = \sum_{i=0}^{6} H_1(c_i) \times CI(c_i) \tag{7}$$

Having applied the core 7-point operator to each of the integral images, we combine the outputs to form a complete edge map. In terms of implementation using the one-dimensional vector structure for the images $I_j, j = 0,...,6$, each output response $D_\lambda^j, j = 0,...,6$ is stored in a one-dimensional vector with non-empty values corresponding to the array positions with indices 0 *mod* 7. These one-dimensional vectors may then be assembled according to the "shifted" structure as illustrated in Figure 6:

$$\forall s_0 \in \{s | s = 0 \, mod \, 7\}, \quad E_\lambda(s_0 + k) = D_\lambda^k(s_0)$$

for $k = 0,...,6$, to yield the consolidated output image $E_\lambda(I_0) = H_\lambda \otimes I_0$ as shown in Figure 7.

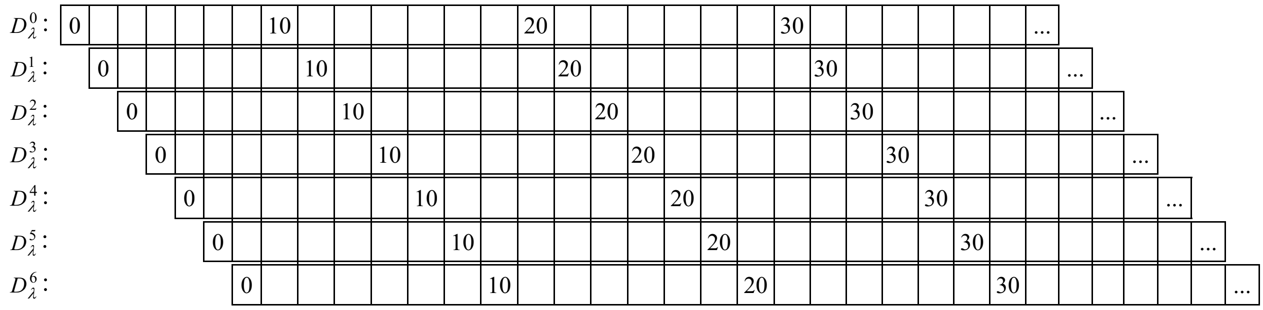


Figure 6. Assembly of the one-dimensional vectors $D_\lambda^j, j = 0,\ldots,6$

$D_\lambda^0(0)$	$D_\lambda^1(0)$	$D_\lambda^2(0)$	$D_\lambda^3(0)$	$D_\lambda^4(0)$	$D_\lambda^5(0)$	$D_\lambda^6(0)$	$D_\lambda^0(10)$	$D_\lambda^1(10)$	$D_\lambda^2(10)$	$D_\lambda^3(10)$	$D_\lambda^4(10)$	$D_\lambda^5(10)$	$D_\lambda^6(10)$	$D_\lambda^0(20)$	…	…

Figure 7. Consolidated output image resulting from assembly of the vectors in Figure 6

5. PERFORMANCE EVALUATION

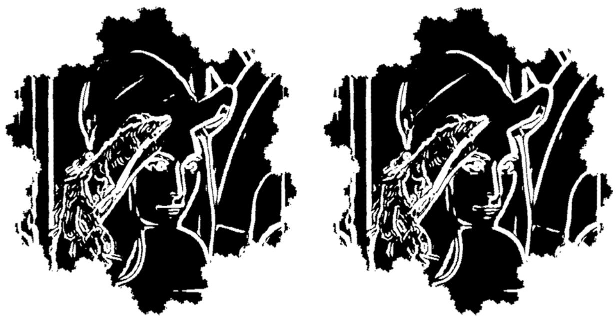
We present results for our proposed hexagonal integral image combined with the biologically motivated approach in [8] in comparison with the original approach in [3], and with standard convolution of an operator with a spiral image where the pixel neighbour addresses are stored in a look-up table (this takes 0.4017s to generate, but is significantly faster than standard hexagonal addressing, which requires *mod* 7 arithmetic). Figure 8 presents edge maps obtained by applying a 49-point operator [3] directly to a standard spiral image and the 7-point operator to hexagonal integral images at scale λ=2. The visual results are similar, whilst there is a notable speed-up in average runtime measured over a set of 10 images, as shown in Table 1. In fact, when using the integral images, applying the 7-point operator at any scale λ will also take only 0.1008 seconds, as each convolution requires only 7 subtractions and 7 multiplications. Hence the technique based on the use of spiral integral images maintains low computational complexity as scale increases. In contrast, 343 multiplications per convolution are required by the other two approaches when λ=3, increasing by a factor of 7 for each increase in scale.

Table 1. Average algorithm run-times

Method	Run-time
Integral Image Approach	0.1008s
Biologically motivated "eye tremor" approach	0.1399s
Spiral convolution using LUT	0.2191s

6. CONCLUSION

We present a novel hexagonal integral image that can be combined with a biologically motivated approach (eye tremor) to speed up feature extraction. We have demonstrated that the approach of applying a core 7-point operator to the spiral integral image at various scales is significantly faster than applying scaled operators to the original image, as we require only 7 subtractions and 7 multiplications to generate each output value regardless of the scale at which the operator is applied.



(a) (b)

Figure 8: (a) 49-point operator applied using Spiral approach; (b) 7-point operator applied using integral eye-tremor approach

7. REFERENCES

[1] Bay H, Tuytelaars T, Van Gool L, "Surf: Speeded up robust features", Proc. ECCV, vol. 1, pp. 404–417, 2006

[2] Bradley D, Roth G, "Adaptive thresholding using the integral image", J. Graphics, GPU, & Game Tools, 12(2), 13-21, 2007

[3] Coleman, SA, Gardiner, B, and Scotney, BW, "Adaptive Tri-Directional Edge Detection Operators based on the Spiral Architecture" IEEE ICIP, pp. 141-144, 2010

[4] Dacey DM, Packer OS, "Receptive Field Structure of h1 Horizontal Cells in Macaque Monkey Retina" Journal of Vision, 2(4), 279-292, 2002

[5] Grabner M, Bischof, H, "Fast Approximated SIFT", Computer Vision ACCV, 918-927, 2006

[6] Middleton L, Sivaswamy J, "Hexagonal Image Processing; A Practical Approach", Springer 2005

[7] Roka A, et al., "Edge Detection Model Based on Involuntary Eye Movements of the Eye-Retina System", Acta Polytechnica Hungarica, 4(1), 31-46, 2007

[8] Scotney, BW, Coleman, SA and Gardiner, B, "Biologically Motivated Feature Extraction Using the Spiral Architecture" IEEE ICIP, pp. 221-224, 2011

[9] Viola P, Jones M, "Rapid object detection using a boosted cascade of simple features", CVPR, vol. 1, pp. 511-518, 2001