# Using Evolution to Learn How to Perform Interest Point Detection

Leonardo Trujillo and Gustavo Olague
Centro de Investigación Científica y de Educación Superior de Ensenada
Departamento de Ciencias de la Computación, División de Física Aplicada, Ensenada B.C. México
trujillo@cicese.mx olague@cicese.mx

## 1. Introduction

The performance of high-level computer vision applications is tightly coupled with the low-level vision operations that are commonly required. Thus, it is advantageous to have low-level feature extractors that are optimal with respect to a desired performance criteria. This paper presents a novel approach that uses Genetic Programming as a learning framework that generates a specific type of low-level feature extractor: *Interest Point Detector*. The learning process is posed as an optimization problem. The optimization criterion is designed to promote the emergence of the detectors' geometric stability under different types of image transformations and global separability between detected points. This concept is represented by the operators repeatability rate [11]. Results prove that our approach is effective at automatically generating low-level feature extractors. This paper presents two different evolved operators: IPGP1 and IPGP2. Their performance is comparable with the Harris [5] operator given their excellent repeatability rate. Furthermore, the learning process was able to rediscover the DET corner detector proposed by Beaudet.

## 2. Related Work

Learning in computer vision is the process in which an artificial system autonomously acquires knowledge from training images to solve a given task. Most published work in this area has centered around solving mid-level and high-level vision tasks. Low-level feature extraction has received less attention. This is primarily due to the fact that common low-level operators have been extensively studied and are well understood. However, evolutionary computation has the ability to endow a learning system with the capability to try new and uncommon image processing strategies that human designers might not consider. Learning Interest Point (IP) operators in this way could provide researchers with deeper insights on the problem domain. Previous work by Ebner [2] poses IP detection as an optimization problem attempting to evolve the Moravec operator [7] using Genetic Programming (GP). The author reports a 15% error compared to Moravec. However, there is not evidence that this is a proper performance metric. A second paper by Ebner [3], presents an evolved operator that is optimized for optical flow estimation. Despite the fact that [3] showed promising results, the optimization criteria used does not guarantee generality for the operator. The work presented in this paper overcomes these limitations by using GP to evolve a computer function that optimizes a standard computer vision performance metric for IP detectors.

## 3. Interest Point Operators

IP detection is a by product of research devoted to corner detection in images. Corner detectors are commonly classified in three main classes: *Contour based methods*, *Parametric model based methods* [9] and *Image intensity based methods* [7, 1, 4, 5]. Corner detectors that operate directly on intensity images are referred to as IP detectors. These operators define a function that extracts a *cornerness measure* for each pixel in an image. This generates a new image known as the *corner response image* and points with the highest *cornerness* response are selected. Points that conform to the geometric concept of a *"corner"*, are not exclusively extracted by common IP operators. They regularly extract points where image intensity variations are high. These type of points are better understood as "*interesting points*". Early IP detectors include Moravec [7] and Beaudet [1]. Extending the work of Moravec, Harris and Stephens [5], and Forstner [4] use the second moment matrix around each point to derive their corner response function. For a more thorough review of interest point and corner detectors we recommend [11] and [9] respectively.

### 3.1. Interest Point Operator Performance

Computer vision applications require local image features that are simple to detect and show geometric stability image transformations. Such transformations include: translation, rotation, illumination change, scale change and affine transformations. Of the previous list of transformations, interest points are only suitable for robust detection in the presence of the first three. Schmid *et. al.* [11] established the repeatability as the quintessential quantification of a detectors performance under different image transformations. However, this measure only characterizes the tractability of interest points and it fails to consider other important and desirable characteristics such as global separability and distinctiveness. Schmid *et. al.* propose a metric for information content around an IP, but fail to generalize it by using local descriptors based on local jets.

### 3.2. Repeatability

A point $x_1$ detected in image $I_1$ is repeated in image $I_i$ if the corresponding point $x_i$ is detected in image $I_i$ [11]. In the case of planar scenes a relation between points $x_1$ and $x_i$ can be established with the homography $H_{1,i}$ where $x_i = H_{1,i}x_1$. The repeatability rate measures the number of repeated points between both images, with respect to the total number of detected points. Both repeated and detected points are only counted if they lie in the common parts of $I_1$ and $I_i$. Detection error is considered by defining that a point is detected at $x_i$ if it lies within a given neighborhood of $x_i$ of size $\epsilon$. The set of point pairs $(x_1^c, x_i^c)$ that lie in the common part of both images and correspond within an error $\epsilon$ is:

$$R_i\left(\epsilon\right) = \left\{\left(x_1^c, x_i^c\right)|dist\left(H_{1,i}x_1^c, x_i^c\right) < \epsilon\right\} \tag{1}$$

The repeatability rate $r_i\left(\epsilon\right)$ at image $I_i$ is:

$$r_i\left(\epsilon\right) = \frac{|R_i\left(\epsilon\right)|}{min\left(\gamma_1, \gamma_i\right)} \tag{2}$$

where $\gamma_1 = |\{x_1^c\}|$ and $\gamma_i = |\{x_i^c\}|$ are the total number of points detected in image $I_1$ and $I_i$ [11].

## 4. Genetic Programming

The modern concept of Genetic Programming (GP) was formalized by John Koza [6]. Koza presented GP as an automated process that used simulated evolution to generate software functions that represent candidate solutions to a given problem. These software functions are represented by *individuals* in a GP *population*. Individuals are coded with a tree representation. Tree nodes contain primitive functions taken from a finite set $F$. Tree leaves contain input elements know as terminals taken from a finite set $T$. Each tree takes terminal elements as input and generates an output according to the node functions. Fitness is assigned to an individual $x$ according to a performance measure $f\left(x\right)$ that compares each individuals output with the desired output that solves a specific task. GP uses standard genetic operators such as mutation and crossover to generate new individuals in the simulated evolutionary process. After a certain termination criteria is met the evolving process is halted and the GP returns the fittest individual found.

## 5. Outline of our Approach

Each individual in the GP population represents a candidate IP operator. Fitness assignment is proportional to its mean repeatability rate $r_J\left(\epsilon\right)$ computed for a set $J = \{I_i\}$ of $n$ training images, where $i = 1...n$. A base image $I_i$ is used to compute the repeatability on all other images in $J$. However, the GP search could easily get lost in unwanted maxima. For example, GP could generate individuals that extract useless points clustered in textureless regions and still have high repeatability rate. Moreover, the training images used have highly textured regions distributed across the image plane. Hence, a *good* detector should extract uniformly distributed points. Consequently, three other terms were incorporated to the fitness function and combined in a multiplicative way:

$$f\left(x\right) = r_J\left(\epsilon\right)\cdot\phi_x^\alpha\cdot\phi_y^\beta\cdot N_\%^\delta \tag{3}$$

where $\phi_x = \dfrac{1}{1+e^{-a(H_x-c)}}$ and $\phi_y = \dfrac{1}{1+e^{-a(H_y-c)}}$ are sigmoidal functions used to promote point dispersion along the $x$ and $y$ directions. The term $H_x = -\sum_x P(x)log_2\left[P(x)\right]$ and $H_y = -\sum_y P(y)log_2\left[P(y)\right]$ represent the entropy value of the spatial distribution of detected interest points along each direction, with $P(x)$ and $P(y)$ approximated by the histogram of interest point localizations. Moreover, because of the logarithmic nature of the entropy function, $\phi_x$ and $\phi_y$ are set to promote entropy values lying in a very small range. The final term $N_\% = \frac{requestedpoints}{extractedpoints}$, is a penalizing factor that reduces the fitness value for detectors that return less than the total number of requested points. Finally, $\alpha$, $\beta$ and $\delta$ control the amount of influence that each term has on $f\left(x\right)$.The function set $F$ contains 6 unary functions and 5 binary functions. All functions input and output are data matrices with the same size as images in $J$. The subset of binary and unary functions are:

$$F_{2ary} = \{+, -, |-|, *, /\} \tag{4}$$

$$F_{1ary} = \left\{A^2, \sqrt{A}, log_2, EQ, G(\sigma=1), G(\sigma=2)\right\} \tag{5}$$

Where EQ is the histogram equalization and $G(\sigma = x)$ are Gaussian filters with blur $\sigma$. The complete function set is:

$$F = F_{2ary} \cup F_{1ary} \tag{6}$$

Great care was taken to design an appropriate terminal set. Thanks to previous understanding of the analytical properties of corner detectors described in [9, 8, 10], we conclude

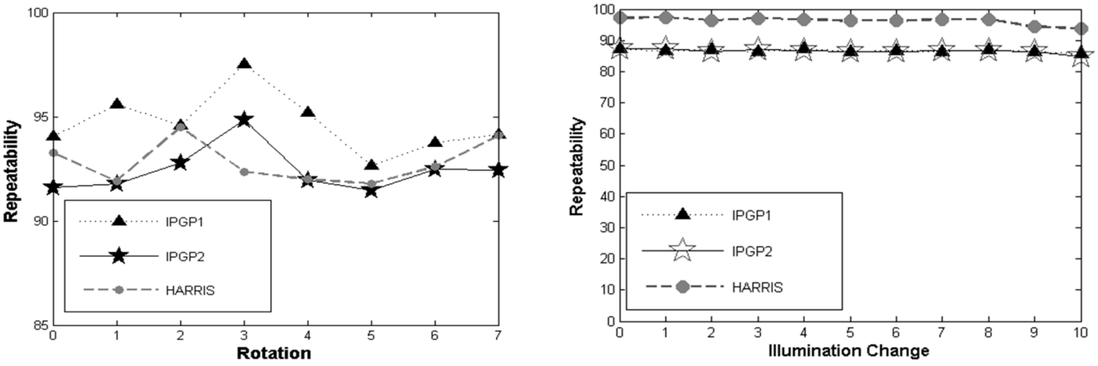

**Figure 1. Performance measures: left) repeatability rate plotted against image rotation for the Van Gogh images; right)repeatability rate plotted against illumination change for the Graph images.**

that an effective IP operator requires information pertaining to the rate of change in image intensity values. Consequently, the terminal set includes first and second order image derivatives. However, we do not claim that this set is necessary nor sufficient and further work will try to determine an optimal set of useful information. Furthermore, the terminal set is image dependent, such that each image $I_i \in J$ has a corresponding $T_i$ defined by:

$$T_i = \{I_i, L_{i,x}, L_{i,x,x}, L_{i,x,y}, L_{i,y,y}, L_{i,y}, I_{i,\sigma=1}\} \qquad (7)$$

Where $L_{i,w} = I_i * G_w(\sigma = 1)$ are image derivatives computed in the $w$ direction with Gaussian kernel derivatives, and $I_{i,\sigma=1}$ is the smoothed image.

### 5.1. Implementation Details

The algorithm was programmed using the Matlab toolbox GPLAB [1]. The image sequence used for training was the Van Gogh set of a planar scene with rotation transformations. For testing, two image sequences were used: Monet and Graph. The former is a sequence of a rotated planar scene and the latter is an image under different illumination conditions. All image sets were downloaded from the Visual Geometry Group website [2], along with Matlab source code to compute the repeatability rate.

## 6. Experimental Results

We present two different interest point operators generated with our approach: IPGP1 and IPGP2 [3]. The first operator IPGP1 has an extremely simple structure. The basic approach of IPGP1 for interest point extraction is blurring the high frequencies of the image. IGP1 does not use image derivatives to extract image regions with high signal variations. It is basically a difference-of-Gaussian operator. A 95% repeatability rate was computed for IPGP1 on the training image set. The second evolved detector IPGP2, represents a modified version of the DET operator proposed by Beaudet [1]. The DET operator is the determinant of the Hessian Matrix around each point. In this instance the GP evolutionary process was able to rediscover a previous man made design. The repeatability rate for IPGP2 on the training set was 92%. Table 1 shows the mathematical expressions for IPGP1 and IPGP2. Figure 1 shows two graphs characterizing the performance of both evolved operators, compared with the Harris detector. Finally, figure 2 shows extracted interest points from samples taken from the three image sequences.

| | |
|---|---|
| IPGP1 | $G(\sigma = 2) * [G(\sigma = 1) * I - I]$ |
| IPGP2 | $G(\sigma = 1) * [L_{xx} \cdot L_{yy}] - G(\sigma = 1) * [L_{xy} \cdot L_{yx}]$ |

**Table 1. Mathematical expressions for IPGP1 and IPGP2.**

## 7. Discussion and Conclusions

This paper presented a novel approach to low-level feature extraction. By posing interest point detection as an optimization problem, we have developed a learning methodology that allows a computer to automatically generate interest point detectors. Learning was conducted with GP and performance was optimized according to the repeatability rate and global separability of extracted points. Experimental results showed that the proposed approach generates re-

[1] http://gplab.sourceforge.net/index.html

[2] http://www.robots.ox.ac.uk/ vgg/research/

[3] IPGP stands for ”Interest Point detector with Genetic Programming”

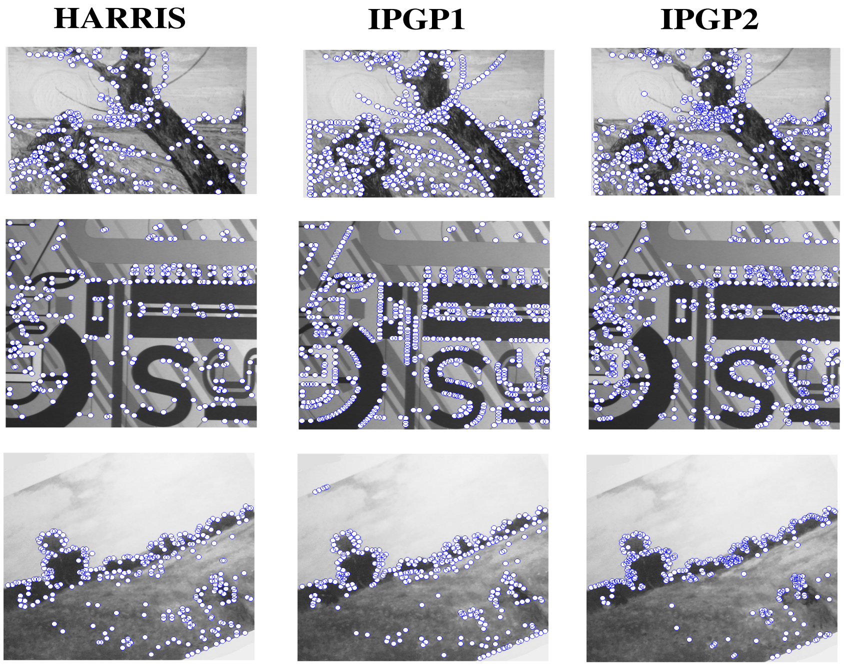

**Figure 2. Sample images of extracted interest points. Columns correspond to Harris, IPGP1 and IPGP2 from left to right. Rows correspond Van Gogh, Graph and Monet from top to bottom.**

liable and compact operators. Furthermore, the experiments conducted displays the abilities of GP to rediscovered a previous man made design, Beaudet's DET operator. Nevertheless, the results presented in this paper, are not intended as a claim that the evolved operators are superior to any other, even if researchers could start to use those proposed here. However, the results clearly demonstrate that learning techniques based on simulated evolution are capable of producing competitive and useful feature extractors. However, much work must still be done in order to determine appropriate optimization criteria if one wishes that effective learning emerges. Moreover, it is our belief that using a similar methodology it will be possible to design learning algorithms that generate appropriate feature extractors for different kinds of computer vision tasks.

**Acknowledgments** This research was funded by UC MEXUS-CONACYT Collaborative Research Grant 2005; through the project "Intelligent Robots for the Exploration of Dynamic Environments". First author supported by scholarship 174785 from CONACyT. This research was also supported by the LAFMI project.

## References

[1] P. R. Beaudet. Rotational invariant image operators. In *Proc. IAPR1978*, pages 579–583, 1978.

[2] M. Ebner. On the evolution of interest operators using genetic programming. In *Proc. EuroGP'98*, pages 6–10, Paris, France, 14-15 1998.

[3] M. Ebner and A. Zell. Evolving a task specific image operator. In *Proc. EvoIASP'99*, volume 1596, pages 74–89, Goteborg, Sweden, 28-29 1999. Springer-Verlag.

[4] W. Forstner. A framework for low level feature extraction. In *Proc. ECCV94*, pages 383–394, Sweden, May 1994.

[5] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. Fourth Alvey Vision Conf.*, volume 15, pages 147–151, 1988.

[6] J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA, 1992.

[7] H. P. Moravec. Towards automatic visual obstacle avoidance. In *IJCAI*, page 584, 1977.

[8] G. Olague and B. Hernández. Flexible model-based multi-corner detector for accurate measurments and recognition. In *Proc. from ICPR2002*, pages 578–583, Quebec, Canada, 2002.

[9] G. Olague and B. Hernández. A new accurate and flexible model based multi-corner detector for measurement and recognition. *Pattern Recognition Letters*, 26(1):27–41, 2005.

[10] G. Olague, B. Hernández, and E. Dunn. Hybrid evolutionary ridge regression approach for high-accurate corner extraction. In *CVPR (1)*, pages 744–749, 2003.

[11] C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of interest point detectors. *Int. J. Comput. Vision*, 37(2):151–172, 2000.