Interest Point Detection through Multiobjective Genetic Programming

Gustavo Olague[a], Leonardo Trujillo[b,*]

[a]*Proyecto Evovisión, Departamento de Ciencias de la Computación, División de Física Aplicada, Centro de Investigación Científica y de Educación Superior de Ensenada, Km. 107 Carretera Tijuana-Ensenada, 22860, Ensenada, BC, México.*
[b]*Instituto Tecnológico de Tijuana, Av. Tecnolgico, Fracc. Tomás Aquino, Tijuana, B.C., México*

Abstract

The detection of stable and informative image points is one of the most important low-level problems in modern computer vision. This paper proposes a multiobjective genetic programming (MO-GP) approach for the automatic synthesis of operators that detect interest points. The proposal is unique for interest point detection because it poses a MO formulation of the point detection problem. The search objectives for the MO-GP search consider three properties that are widely expressed as desirable for an interest point detector, these are: (1) stability; (2) point dispersion; and (3) high information content. The results suggest that the point detection task is a MO problem, and that different operators can provide different trade-offs among the objectives. In fact, MO-GP is able to find several sets of Pareto optimal operators, whose performance is validated on standardized procedures including an extensive test with 500 images; as a result, we could say that all solutions found by the system dominate previously man-made detectors in the Pareto sense. In conclusion, the MO formulation of the interest point detection problem provides the appropriate framework for the automatic design of image operators that achieve interesting trade-offs between relevant performance criteria that are meaningful for a variety of vision tasks.

Keywords: Multiobjective optimization, genetic programming, interest point detection, evolutionary computer vision.

*Corresponding author
Email addresses: olague@cicese.mx (Gustavo Olague), leonardo.trujillo.ttl@gmail.com (Leonardo Trujillo)

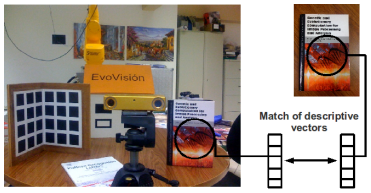


Figure 1: The matching process between two images using local features and descriptors. A local descriptor is computed for an interest region detected on the left image. Then, a correspondence is sought with the local descriptors extracted from the image on the right. A correct match between both vectors implies a match between the corresponding regions.

1. Introduction

Over the past decade, many researchers have developed computer vision systems that base their core functionality on the detection and description of sparse local features. The basic approach was introduced in [1, 2], and consists of the following steps. First, small image regions centered around salient image pixels, better known as *interest points*, are detected [3, 4]. Then, each region is described using compact numerical vectors that attempt to capture the main characteristics of local shape and appearance; these are called *local image descriptors* [5, 6, 7]. The set of local regions and their corresponding descriptors are used to construct models of objects or the scene captured within the image. Afterwards, when a new image is analyzed, this process is repeated and the extracted features are compared with the stored models. In this manner, a vision system can search for correspondences of local features; a simple example is depicted in Figure 1. This approach provides several advantages for high-level systems: (1) traditional image segmentation, a difficult mid-level task, is not required; (2) the approach is robust to partial occlusions and to several types of geometric and photometric transformations; and (3) the total amount of image information is sharply reduced because only a subset of image regions are analyzed and described by a compact representation.

The performance of vision systems that are based on this approach will depend on the quality of the algorithms used to detect and describe the local features. In both cases, many proposals have been developed over the past thirty years, and most of them have been derived using standard techniques from computer vision research [8, 3]. Recently, however, some researchers have developed Genetic Programming (GP) algorithms for the automatic design of image operators that can detect [9, 10, 11, 12, 13, 14, 15] and describe [16, 17] local image features. The underlying hypothesis of these GP-based proposals is that current methods should not be considered to be

optimal in any general sense and that the design of interest point detection can be accomplished through a machine learning approach based on genetic programming. Therefore, these works have proposed to use evolutionary algorithms to search for specialized image operators that can outperform the standard techniques.

Following this line of research, this paper presents a multiobjective GP (MO-GP) approach to design interest point operators. The main contribution is that the problem of interest point detection is formulated in terms of MO optimization, which is unique. Moreover, we show that MO-GP can produce a diverse set of detectors that achieve different trade-offs between the various performance criteria that are widely used to evaluate interest point detectors. Considering the problem domain, this paper is closely related to other GP applications in computer vision, of which several examples are reviewed next.

1.1. Genetic Programming and Computer Vision

In computer vision, most problems are often ill-defined and considered to be intrinsically very difficult. Therefore, some researchers have attempted to reduce the complexity of designing vision systems by relying on automated methods, such as evolutionary computation [18]. In particular, GP has proven to be a useful paradigm in the development of vision algorithms because of its ability to directly generate specialized computer functions or mathematical operators [19, 20]. For instance, some have used GP to search for operators that can detect low-level features that were predefined by human experts, features that posses a clear semantic interpretation such as corners [9] or edges [21]. Also, GP has achieved promising results in measuring a more formal characteristic image structure, the Hölderian regularity [22]. Others have used GP to construct domain-specific features that need not be interpretable by a human expert [19, 23, 24]. On the other hand, GP has been used to directly solve a specific visual task, such as image segmentation [25], object recognition and detection [26, 27], texture analysis [28] and content based retrieval [29]. Other recent examples are [16, 17] where GP is used to optimize a part of the SIFT algorithm [2], the most widely used method for region description.

However, all the above cited works pose the problem of evolving vision algorithms in terms of a single objective function. Therefore, despite their promising results, they are limited by the manner in which other objective criteria could be added to help guide the search, especially if conflicts exist among them. Moreover, the evolutionary process, which is normally computationally

expensive, needs to be executed several times in order to possibly obtain more than a single type of solution. One manner in which these difficulties could be lessened, or eliminated, is to use a MO approach based on Pareto dominance relations. However, MO search techniques have not gained a wide acceptance within the vision community and only a small number of examples exist. For instance, in [30] a MO genetic algorithm is used to solve the sensor planning problem for vision metrology. Other examples include [31] where MO-GP is used to automatically learn simple visual concepts, and [32] where MO-GP is used to control program size for a feature extraction problem. More recently, in [15] we present a hybrid computer assisted design approach to develop a feature detection algorithm using genetic programming.

This paper presents a MO approach to synthesize image operators that detect interest points. The MO formulation allows us to integrate several performance criteria in a principled manner, and produce a non- dominated set of near-optimal solutions. The evolved operators provide different trade-offs among the various objectives, that are based on standard evaluation criteria. Moreover, experimental results validate the approach and illustrate the intrinsic MO nature of the interest point detection problem.

This work is a continuation of previous contributions we have developed on this subject. In [11, 14] interest point operators were synthesized by considering a single objective, and in [13, 15] only preliminary results of the MO approach were presented. Therefore, in order to contextualize our work, we list the main contributions made in this paper.

1.2. Main Contributions

- This paper studies the problem of interest point detection from the standpoint of MO optimization, an extension of the single objective approach followed in [11, 14] and a continuation of preliminary results published in [13, 15]. The proposal is unique in the MO treatment of this ubiquitous computer vision task providing novel insights for future research.

- The MO-GP algorithm considers Pareto dominance relations and produces several sets of near-optimal operators. Three well-known performance criteria are considered: (1) Stability; (2) Point Dispersion; and (3) Information Content. Moreover, the MOGP can integrate more objectives in a straightforward and principled manner.

- Finally, several novel detectors are presented, each exhibiting different trade-offs between each of the considered objectives. These operators provide novel tools for vision applications

that outperform, in the Pareto sense, many previously proposed methods. Moreover, the performance of the evolved operators is validated with an extensive database of 500 test images of real world scenes.

The remainder of this paper is organized as follows. Section 2 presents an introduction to interest point detection, a review of previous proposals, and a description of the performance criteria used to evaluate them. Afterwards, Section 3 provides a brief overview of GP. In Section 4, we describe how the problem of interest point detection can be solved through GP. In particular, we review our previous single objective proposal and introduce the multiobjective approach. Then, Section 5 presents the experimental setup and results. Finally, a summary and concluding remarks are given in Section 6.

2. Interest point detection

Interest points are simple point features, image pixels that are salient or unique when compared with neighboring pixels. These features are used by a wide range of applications in image analysis, computer vision and photogrammetry [3, 33]. The algorithms used to detect interest points analyze the intensity patterns within local image regions and only make weak assumptions regarding the underlying structure [8]. For instance, another type of point feature are corners; however, these points have a clear geometric interpretation that assumes the intersection of two borders or lines [34]. Interest points are much more general and often lack a semantic interpretation. It suffices to say that interest points are quantitatively and qualitatively different from other points, and that they usually represent only a small fraction of the total number of image pixels. In other words, interest points can be more usefully defined based on a set of objective properties that they fulfill or through their measurable impact on higher level tasks.

2.1. Problem Definition

A measure of how salient or interesting each pixel is can be obtained using a mapping of the form $K(\mathbf{x}) : \mathbb{R}^+ \rightarrow \mathbb{R}$ called an interest point operator. Thus, each interest point detector employs a different operator K; in this way, a *detector* refers to the complete algorithmic process that extracts interest points, while an *operator* only computes the interest measure. Applying K to an image I produces what can be called an *interest image* I^*, see Figure 2. Afterwards, most detectors follow

Figure 2: A look at interest point detection: left, an input image I; middle, *interest image* I^*; right, detected points superimposed on I after non-maximum suppression and thresholding.

the same basic process: non-maxima suppression that eliminates pixels that are not local maxima, and a thresholding step that obtains the final set of points. Therefore, a pixel $\mathbf{x}$ is tagged as an interest point if the following conditions hold,

$$K(\mathbf{x}) > max\left\{K(\mathbf{x_W})|\forall \mathbf{x_W} \in \mathbf{W}, \mathbf{x_W} \neq \mathbf{x}\right\} \wedge K(\mathbf{x}) > h\,, \tag{1}$$

where $\mathbf{W}$ is a square neighborhood of size $n \times n$ around $\mathbf{x}$, and h is an empirically defined threshold. The first condition in Eq. 1 accounts for non-maximum suppression and the second is the thresholding step, see Figure 2. In this work $n = 5$ and h is operator dependant.

Therefore, the problem of interest point detection, as stated above, is that of defining K. The question then is, what should K be? The answer will depend on the type of information that a vision system needs to extract from an image to accomplish a high level task. Nevertheless, notice that the above definition does not enforce any restrictions regarding the underlying structure or appearance of the image, but it is only based on the functional behavior of K. This contrasts with the semantic definition that can be given to borders, corners or line junctions. Therefore, after reviewing previous proposals for point detection below, we focus on three principles: repeatability, point distribution, and information content; that researchers in computer vision have proposed to quantify the overall quality or usefulness of an interest point detector. In particular, we applied objective criteria using the well established repeatability measure, and we developed a point distribution measure based on image entropy; as well as, an information content criterion based on the Lipschitz exponent. Afterwards, our approach will be to use these criteria as search objectives for an automated optimization process to synthesize novel operators for point detection.

2.2. Previous Work

The problem of detecting interest points has been well-studied and a large variety of proposals exist in current literature. For instance, the most widely used methods employ image operators

that are based on the local second-moment matrix, defined as

$$A(\mathbf{x}, \sigma_I, \sigma_D) = \sigma_D^2 \cdot G_{\sigma_I} * \begin{bmatrix} L_x^2 & L_x L_y \\ L_x L_y & L_y^2 \end{bmatrix},$$

where σ_D and σ_I are the differentiation and integration scales respectively, and $L_u = L_u(x, \sigma_D)$ is the Gaussian derivative in direction u of image I at point $\mathbf{x}$. Some researchers have argued that A is an optimal model for point detection [35], and many detectors extract an interest measure using it[36, 37], including the most widely used detector by Harris and Stephens [38],

$$K_{Harris\&Stephens}(\mathbf{x}) = det(A) - k \cdot Tr(A)^2 \,. \tag{2}$$

Other detectors employ operators that extract measures that are related to the local curvature at each point. For instance, [39] uses the determinant of the Hessian matrix and in [40] the operator uses the gradient magnitude and the intensity with which the gradient changes direction. These operators, and others, have been shown to be mathematically equivalent [41]. The interested reader, is referred to [3] for an extensive review.

2.3. Performance Evaluation

The large number of proposed interest point detectors makes it difficult to determine which is best suited for a particular task [3]. Therefore, researchers have developed objective evaluation criteria that can be used to make meaningful comparisons between various detectors. In summary, two approaches have been taken: (1) based on analytical properties of the detectors, and (2) using experimental criteria. An example of the former approach is [35], where the authors propose a set of mathematical axioms that an optimal detector is expected to satisfy. Their approach is based on the assumption that the second-moment matrix is an essential component for interest point detection which the authors justify by only considering the detection of well-defined corners. However, the concept of an interest point is much more general since in most real-world scenes corner points can be quite rare. Indeed, attempting to analytically describe the optimal performance of an interest point detector is very difficult because of the large variety of image structures that are associated with interest points.

Therefore, most methods of evaluation follow an experimental approach, which can be done in different ways. One possibility is to evaluate a detector based on the performance it achieves

as a component of a higher-level system. For instance, [10] evaluates detectors based on the performance of an optical flow estimation algorithm. As a result, any conclusions that are drawn from such an application-specific evaluation will be quite limited in scope, and could not be used to infer its expected performance on a new task. Therefore, here we follow [33, 8, 42, 43], and employ general experimental criteria that attempt to capture the basic properties that are normally expected from a set of detected interest points when used in a variety of vision tasks, such as uniqueness, stability and distinctiveness [33]. In what follows, we describe three of the most common criteria that have been proposed.

2.4. Stability

The stability of a detector is paramount for applications where the same image feature must be tracked across multiple images or when the feature needs to be detected on images taken from different viewpoints. The best measure for the stability of a detector is the repeatability rate, which provides an estimate of how independent the detection process is with respect to conditions under which the image is acquired [8]. In fact, the repeatability as a measure of invariance became the standard approach towards evaluating point detectors in computer vision literature.

The repeatability rate is computed as follows. An interest point $\mathbf{x}_1$ detected within image I_1 is said to be repeated in image I_i if the corresponding point $\mathbf{x}_i$ is detected in image I_i. In the case of planar scenes, a relation between points $\mathbf{x}_1$ and $\mathbf{x}_i$ can be established using a planar homography $H_{1,i}$, where $\mathbf{x}_i = H_{1,i}\mathbf{x}_1$, see Figure 3. Therefore, the repeatability rate measures the number of repeated points between both images relative to the total number of detected points.

A repeated point is said to be detected at pixel $\mathbf{x}_i$ if it lies within a given neighborhood of size $\epsilon = 1.5$ pixels. The set of point pairs $(\mathbf{x}_1^c, \mathbf{x}_i^c)$ that lie in the common part of both images and correspond within an error ϵ is defined as

$$R_{I_i}(\epsilon) = \left\{(\mathbf{x}_1^c, \mathbf{x}_i^c) \,|dist\left(H_{1,i}\mathbf{x}_1^c, \mathbf{x}_i^c\right) < \epsilon\right\} \ . \tag{3}$$

Thus, the repeatability rate $r_{I_i}(\epsilon)$ of the detected points from image I_i with respect to the points from image I_1, is given by

$$r_{I_i}(\epsilon) = \frac{|R_{I_i}(\epsilon)|}{min\left(\gamma_1, \gamma_i\right)} \ , \tag{4}$$

where $\gamma_1 = |\{\mathbf{x}_1^c\}|$ and $\gamma_i = |\{\mathbf{x}_i^c\}|$ are the total numbers of points extracted from image I_1 and image I_i respectively.

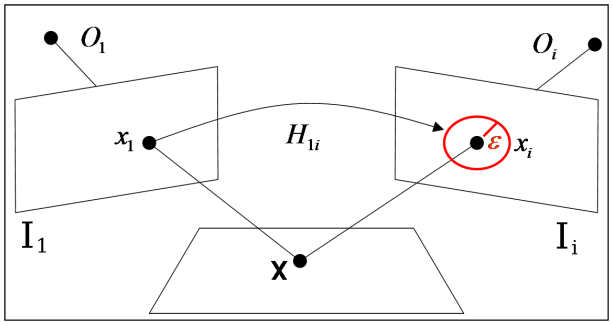


Figure 3: A 3D point is projected onto points $\mathbf{x}_1$ and $\mathbf{x}_i$ on images I_1 and I_i respectively. $\mathbf{x}_1$ is said to be repeated by $\mathbf{x}_i$, if a point is detected within a neighborhood of $\mathbf{x}_i$ of size ϵ. For planar scenes $\mathbf{x}_1$ and $\mathbf{x}_i$ are related by the homography $H_{1,i}$ [8].

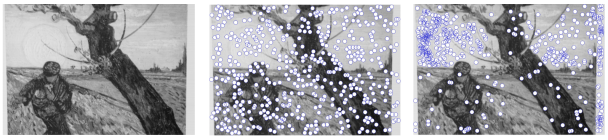


Figure 4: Point Dispersion: Original image (left), highly dispersed points (center), small amount of dispersion (right).

2.5. Point Dispersion

Another performance criterion is to consider the amount of dispersion that the interest points have over the image plane, see Figure 4. The method described here was originally proposed in [11]. It is normally expected that a set of highly dispersed points will provide a better sampling of the information contained within an image and provide more useful data for higher levels of analysis. Although this criterion will greatly depend upon the underlying structure of the imaged scene, some authors have stated that it is an important determining factor when choosing a method for point detection in specific domains [33, 44, 45].

A measure for point dispersion can be obtained by using the entropy computed from the partition $\{\mathcal{I}_j\}$ of the plane of image I; where $\{\mathcal{I}_j\}$ represents a grid over I, and each $\mathcal{I}_j$ describes an individual bin. Hence, $\mathcal{D}$ is the entropy value of the spatial distribution of detected interest points given by

$$\mathcal{D}(I, X) = -\sum P_j \cdot log_2(P_j) \,, \tag{5}$$

where P_j is approximated by the 2D histogram of the position of interest points within I. In this work, the image is divided into a 2D grid with a bin size of 8×8 pixels [13].

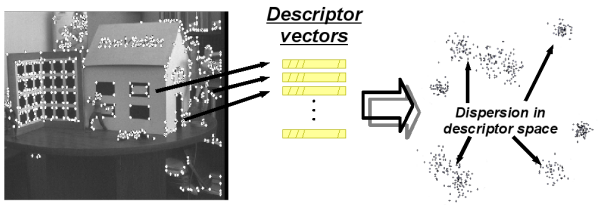


Figure 5: It is desirable to detect interest points that have unique descriptions. Hence, the set of corresponding descriptors should be dispersed within the space of possible descriptors.

2.6. Information Content

As stated earlier, interest points are normally used in conjunction with a descriptor that characterizes local shape and appearance. In this scenario, the local region should be as unique as possible; i.e., the set of local descriptors should be highly dispersed within the space of possible descriptors, see Figure 5. The uniqueness of each descriptor can increase the discriminatory power of the models that are built with them. This would lead to an easier matching process with a low number of false correspondences and a higher number of correct matches, by providing more discriminant descriptors and thus improving the performance of the vision system. Conversely, if interest points are located on similar image features then their descriptors can be expected to be homogeneous and thus convey less information.

The concept of the information content extracted from a set of interest points was introduced by [8] and described as the relative likelihood of computing a local descriptor at any given point. For every interest point $\mathbf{x}$ a corresponding local image descriptor γ is computed. Therefore, if we consider that a detector identifies a set X of n interest points, there will be a corresponding set of descriptors Γ, where $\forall\ \mathbf{x} \in X\ \exists\ \gamma \in \Gamma$. Moreover, let Υ represent the space of all possible descriptors. If the descriptors in Γ are crowded within a small region of Υ, then the set X conveys a small amount of information, denoted by $\mathcal{I}$, with the converse being true in the opposite case. Based on information theory, $\mathcal{I}$ can be obtained by using the amount of entropy contained within the set of the descriptors Γ. Therefore, if we consider a partition $\Upsilon = \{\Upsilon_j\}$, and the probability q_j is approximated by the histogram of the descriptors $\gamma \in \Upsilon_j$ from set Γ, then the information content of X can be defined as

$$\mathcal{I}(\Gamma) = -\sum q_j \cdot log_2(q_j)\ . \tag{6}$$

To compute the information content we must first choose which local descriptor to use. In [8]

the authors used the local jet around each point as their descriptive vector [46]. However, a more recent experimental comparison showed that the local jet does not provide a good characterization of local image information [5]. Results suggested that the Scale Invariant Feature Transform (SIFT) [47] extracts the most discriminative set of descriptors. Therefore, in preliminary work with the MO-GP approach we have used the SIFT descriptor to measure information content [13]. However, the manner in which SIFT builds its descriptive vector leads the evolutionary search towards counterintuitive results, such as those shown in [13], where highly cluttered points appear to have a large variety of associated descriptors. The problem arises because SIFT builds an histogram of the gradient orientations within each region. Therefore, when an image region contains a curved or circular shape the SIFT descriptors might change drastically between neighboring points within the region. In this scenario, it is possible to extract a varied set of descriptive vectors from a set of points that are clustered together within the same region. Hence, even if two overlapping regions exhibit a similar appearance their corresponding descriptors might still be quite different, a counterintuitive result. In practice, however, this is normally not a problem for SIFT because it was intended to be used in conjunction with a traditional detector that extracts points for which the gradient magnitude reaches a local maximum in both principal directions. Therefore, the scenario described above is not expected in normal usage of the SIFT descriptor. However, the GP search can generate operators that do not use the same detection criteria so there is no limit to what the GP might attempt to try during evolution. We have therefore concluded that computing the information content with SIFT can lead the search towards determining that points that are spatially very close can still be described in very different ways. Therefore, in the present work we use a descriptor based on the pointwise Hölder exponent, a measure of image regularity at each point [6, 48].

2.6.1. Hölder Descriptor

Most of the useful information contained within a signal is located within the irregular or singular regions. Hölderian regularity provides a characterization of such singular structures [49], and can be quantified by the pointwise Hölder exponent.

Definition 1:Let $f : \mathbb{R} \to \mathbb{R}$, $s \in \mathbb{R}^{+*} \setminus \mathbb{N}$ and $x_0 \in \mathbb{R}$. $f \in C^s(x_0)$ if and only if $\exists \eta \in \mathbb{R}^{+*}$, and a polynomial P of degree $< s$ and a constant c such that

$$\forall x \in B(x_0, \eta), |f(x) - P(x - x_0)| \leq c|x - x_0|^s \ , \tag{7}$$

where $B(x_0, \eta)$ is the local neighborhood around x_0 with a radius η [1]. The pointwise Hölder exponent of f at x_0 is $\alpha_p(x_0) = \sup_s \{f \in C^s(x_0)\}$.

Hölderian regularity refines the concept of the Taylor series approximation of a function by also accounting for non-differentiable points [49]. For most signals, the Hölder exponent cannot be computed analytically, but it can be estimated using a variety of methods, including the oscillations method [50] or even using an evolved estimator [22].

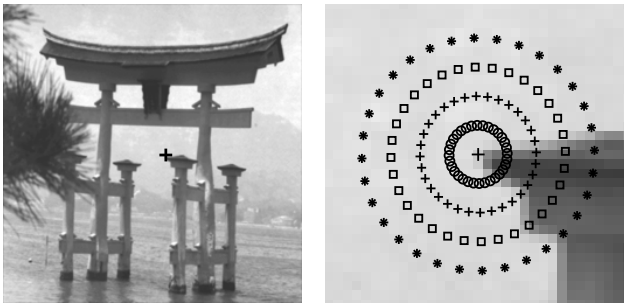
A local descriptor based on Hölderian regularity can be constructed very simply [6]. The idea is to uniformly sample the Hölder exponent using a circular grid within each region. For instance, Figure 6a shows an interest point detected within a test image, and Figure 6b presents the local region around it. The descriptor is then constructed by sampling the exponent of the central point and at 32 equidistant points at four different radii, this gives a vector dimension of 129; Figure 6b illustrates this process. The Hölder descriptor has two useful properties for region description. First, because the exponent is estimated using oscillations which are relative intensity differences within the region, there is no need to normalize the descriptor for uniform intensity variations. Second, rotation invariance can be obtained by ordering the values in the descriptor based on the principal orientation of the gradient within each region.

The local Hölder descriptor provides a better characterization of local features for the optimization problem we have posed because it is able to capture the main similarities of neighboring pixels within the image. Moreover, it does so without sacrificing a distinctive description of salient points, given the comparable performance it achieves when compared to SIFT on standard matching tests [6, 48] .

3. Overview of Genetic Programming

The field of evolutionary computation focuses on the development of search and optimization algorithms that are based on the core principles of Neo-Darwinian evolutionary theory [51]. Evolutionary algorithms are population-based meta-heuristics, where candidate solutions are stochastically selected and modified to produce new, and possibly better, solutions for a particular problem. The selection process favors individuals that exhibit the best performance, and the process is carried out iteratively until a termination criterion is reached, such as a maximum number of

[1] $\mathbb{R}^{+*} \setminus \mathbb{N}$ stands for the set of positive non-zero real numbers excluding the natural numbers.



(a) Interest point (b) Sample points

Figure 6: The local Hölder descriptor for salient image regions.

iterations (generations). Of the current algorithms, GP represents one of the most advanced forms of evolutionary search and optimization, originally developed in its modern form by John Koza [52]. In canonical GP, each individual is encoded using a tree structure, also known as a program tree, which can express a simple computer program, function, or operator. Individual trees are built using elements from two finite sets, internal nodes contain basic functions from a *Function* set F, and leaves contain the input variables of the program, taken from *Terminal* set T. These sets define the search space for a GP system, which is normally very large, and could be infinite if a strict limit on size or depth is not enforced during the run for each tree. In most scenarios, when using a tree-based GP there are two main aspects that must be defined: (1) the search space, sets F and T; and (2) the objective function or fitness function.

The basic GP algorithm contains a single evolutionary loop, as any evolutionary algorithm, with three main stages: fitness evaluation, fitness-based selection, and the production of new program trees by genetic operators. The main loop is similar to the more widely known genetic algorithm (GA), where the main difference lies in the manner in which individuals are coded. While a GA uses bit strings, or parameter vectors, with a constant and uniform length among all individuals, GP uses tree structures which can be of variable size. As a result, another important difference is the manner in which new program trees are constructed. There are two main operations. First, crossover effectively swaps two randomly selected subtrees between two trees that were chosen based on their fitness. Crossover works under the assumption that if two individual trees have a good fitness value, then combining both might produce program trees of even better fitness. The second operator is mutation, where a randomly chosen subtree is deleted and substi-

tuted by a new random tree. The role of mutation is to introduce diversity into the population and to possibly provide slight fitness improvements.

4. Interest point detection with Genetic Programming

4.1. Previous Work: Single Objective Approaches

The first attempts to synthesize interest point operators with GP were presented by Ebner and colleagues [9, 10]. In [9], the goal was to evolve an operator that would detect the same points that are detected with the Moravec detector [53]. Then, in [10] the fitness function was based on the performance of a high-level system that computes the optical flow for a single image sequence. These proposals did produce promising results, but both works suffer from the fact that they fail to incorporate a proper measure of performance. In the former case, the Moravec detector is not a competitive detector, if compared with state-of-the-art methods. In the latter case, the authors were primarily concerned with solving a specific problem instance, instead of evolving operators that could represent a more general solution for interest point detection.

In [11, 14], we defined a fitness function that multiplicatively combines the repeatability rate with the amount of point dispersion exhibited by the set of detected points. The former criterion is the principle guide for the evolutionary process, while the latter serves as a constraint on the lower and upper bounds of what was considered to be an adequate amount of dispersion. The results achieved with this method produced competitive operators that equaled or outperformed previous man-made detectors. In fact, some of the evolved operators have been easily extended to the case of scale invariant detection [12]. A noteworthy observation was that GP did not evolve operators that incorporated elements from the second-moment matrix, contradicting some previous assumptions [35]. We conjecture that operators that rely on the second-moment matrix are located on isolated peaks of the fitness landscape and are therefore quite difficult for GP to reproduce.

The single objective approach produced good results but it is hampered by several limitations. First, the objective function includes several parameters that require a precise tuning [14]. Second, it is not clear how to incorporate other criteria within the objective function; for example, adding a term that promotes a high information content. Finally, canonical single objective evolution tends to produce convergence within the population and thus generate a single type of solution. This is particularly prohibitive for our algorithm because evolution required approximately 24 hrs of

computation time per run [54] [2]. Therefore, a plausible improvement is to employ an optimization algorithm that uses straightforward objective functions, facilitates the combination of multiple objectives, and maximizes computational resources by producing several operators from a single run.

4.2. Multiobjective Approach

In order to overcome the shortcomings of the single objective approach described above, we propose a multiobjective formulation of the interest point detection problem [13]. A MO approach allows us to incorporate several optimization criteria without the need of tuning a parameterized objective function by combining criteria in an ad-hoc manner. Moreover, MO optimization provides a principled manner in which to add new objectives without significantly altering the search algorithm. Finally, MO algorithms search for a set of Pareto optimal solutions instead of a single global optimum. Hence, by definition these algorithms return a variety of near-optimal solutions (non-dominated set), each exhibiting a different trade-off between the optimization objectives.

4.2.1. Multiobjective Evolutionary Algorithms

Optimization based on multiple objectives dates back to the seminal work on economic theory developed by Pareto [55]. As a result, for many years scientific discourse related to such methods was mostly limited to the fields of operations research and economics. Today, multiobjective optimization is reaching new research areas thanks in great part to the wide scope of evolutionary computing. Multiobjective optimization is considerably more complex than single criterion optimization and is considered as a separate field of research. The principal difference resides with how the concept of optimality is defined. In contrast with single objective optimization where the concept of optimality is trivially defined in a monodimensional space; in multiobjective optimization optimality is based on dominance relations among solutions being evaluated in a multidimensional space.

Therefore, in MO optimization a decision maker considers two different and complimentary spaces: one for decision variables and another for the objective functions. For real valued functions the two spaces are related by the mapping $\vec{f} : \mathcal{R}^n \rightarrow \mathcal{R}^k$. The set of constraints on $\vec{f}(\mathbf{x}) =$

[2] Note that the computation time is related to the GP search; nevertheless, the final interest point operator is simple and computationally fast. This is explored further at the end of the experimental section.

$[f_1(x), ..., f_k(x)]$ define a feasible region $\Omega \subset \mathcal{R}^n$ in decision space along with its corresponding image $\Lambda \subset \mathcal{R}^n$ within objective function space. The optimum is found on a frontier of the objective space, which is called the *Pareto Front*, while the corresponding decision variables in Ω are called the *Pareto-Optimal Set*. As stated above, optimality is based on *Pareto Dominance* relations among different solutions, which are specified in objective space as follows. An objective vector $\vec{f^i}$ is said to dominate another objective vector $\vec{f^j}$, $\vec{f^i} \succ \vec{f^j}$, if no component of $\vec{f^i}$ is larger (considering a minimization problem) than the corresponding component of $\vec{f^j}$, and at least one component is smaller.

The goal of multiobjective optimization is to obtain a set of multiple solutions, all of which are optimal in the Pareto sense. Such a set of solutions is based on dominance criteria; if the objectives are in conflict then a single optimal solution cannot exist. Moreover, in real world situations most MO problems lack a closed form solution hampered by several objectives and constraints. Therefore, there is a need for useful computational search methods that can succeed in obtaining an approximation of the Pareto-optimal set. An important aspect is to achieve a good performance and this is measured through the satisfaction of two important criteria. First, the search should converge towards the true Pareto Front, the analog of the optimal point for a single objective. Achieving this is difficult when the objective landscape is discontinuous or irregular. Second, the search must representatively sample the true Pareto Front with a diverse set of solutions. Depending on the structure of the objective space, however, some regions of the true Pareto Front may not be reachable by the search process.

Research on evolutionary computing about how to solve MO problems has produced a variety of search-based approaches which are widely known as MO evolutionary algorithms (MOEAs). These algorithms have proven to be a powerful paradigm in MO optimization and many different proposals have been developed [56, 57]. It is possible to summarize three main properties shared by modern MOEAs that are useful in achieving good performance. First, fitness assignment considers Pareto dominance to bias the search towards the Pareto Front. Second, because a uniform sampling of the Pareto Front is desired, current MOEAs use diversity preservation within the objective space. Finally, MOEAs rely on elaborate survival strategies that use elitism through a population archive that stores the algorithm progression.

In this work, we have chosen the Strength Pareto Evolutionary Algorithm 2 (SPEA2) [58] within our proposed MO-GP search. SPEA2 is a third generation MOEA that accounts for dominance and

non-dominance relations between the individuals in the current population and individuals in the population archive. In SPEA2, each individual i is assigned a strength value $S(i)$ that represents the number of solutions it dominates. On the basis of the S values the raw fitness $R(i)$ of an individual is determined by the strengths of its dominators in both the archive and the population. Note that R is to be minimized with this formulation; see [58] for further technical details.

Diversity preservation is achieved by using a *k-th* nearest neighbor clustering algorithm that penalizes individuals that reside in densely populated regions of objective space. The density $D(i)$ of an individual i is given by $D = \frac{1}{{d_i}^k + 2}$, where ${d_i}^k$ is the distance within objective space of i to its $k-th$ nearest neighbor and k is set to 1. With this formulation the density is bounded by above $D(i) < 1$ and it is used as a penalization factor added to the raw fitness $R(i)$. Therefore, the fitness of an individual i is given by

$$F(i) = R(i) + D(i) \; . \tag{8}$$

Moreover, SPEA2 uses a fixed-size archiving approach and a truncation scheme that promotes diversity by removing individuals that have the minimum distance to its neighbors. Finally, it preserves boundary solutions by using a carefully designed selection operator.

4.2.2. Search Space

The search space we propose is based on basic image operations that are normally used by several of the previously proposed interest point operators that have been extensively used. Therefore, these sets are defined as in [14], given by

$$\begin{aligned} F = \left\{+, |+|, -, |-|, |I_{out}|, *, \div, I_{out}^2, \sqrt{I_{out}}, log_2(I_{out})\right\} \\ \bigcup \left\{k \cdot I_{out}, \tfrac{\delta}{\delta x} G_{\sigma_D}, \tfrac{\delta}{\delta y} G_{\sigma_D}, G_{\sigma=1}, G_{\sigma=2}\right\} \; , \\ T = \left\{I, L_x, L_{xx}, L_{xy}, L_{yy}, L_y\right\} \; , \end{aligned}$$

where I is the input image, and I_{out} is any of the terminals in T or the output of any of the functions in F; L_u is a Gaussian image derivative along direction u; G_σ are Gaussian smoothing filters; $\frac{\delta}{\delta u} G_{\sigma_D}$ represents the derivative of a Gaussian function; and a scale factor of $k = 0.05$. Here, as in [14], we do not claim that these sets are either necessary nor sufficient. However, they do incorporate the most common basic operations that are commonly used by point detection algorithms, and others as well. In fact, this gives the GP the necessary building blocks to construct previously proposed detectors, and they are amenable to an incremental optimization process. Moreover, these basic elements achieved good results with the single objective GP [14].

4.2.3. Objective Functions

In Section 2.3 we introduced three performance measures for interest point detectors, from which we must derive three objective functions which are here constructed as cost functions that must be minimized. As stated above, in [14] we proposed a single objective function that scales linearly with the repeatability rate $r_{K,J}$. In preliminary tests, similar objective functions based on $r_{K,J}$, $\mathcal{D}(I,X)$ and $\mathcal{I}_\mu$ were tested. However, the SPEA2 search produced Pareto fronts that were poorly covered by the evolved operators. Therefore, we propose the following cost functions for stability $f_1(K)$, dispersion $f_2(K)$, and information content $f_3(K)$; given by:

$$f_1(K) = \frac{1}{r_{K,J}\left(\epsilon\right) + c_1} \, . \tag{9}$$

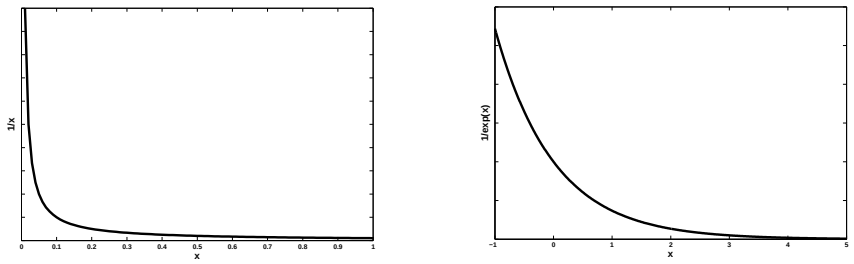
$$f_2(K) = \frac{1}{\exp(\mathcal{D}(I,X) - c_2)} \, . \tag{10}$$

$$f_3(K) = \frac{1}{\exp(\mathcal{I}_\mu(\Gamma) - c_3)} \, . \tag{11}$$

where K represents an individual operator within the GP population. For $f_1(K)$, $r_{K,J}(\epsilon)$ represents the mean repeatability rate computed with a sequence J of progressively transformed images, where a repeatability score is obtained between a base image and each of the other images in J. The constant is set to $c_1 = 0.01$ to protect against a division by zero. In this case, the range of $r_{K,J}(\epsilon)$ is $[0,1]$ and f_1 is the reciprocal function with the form depicted in Figure7(a).

In f_2 and f_3, $\mathcal{D}(I,X)$ and $\mathcal{I}_\mu$ respectively represent the amount of point dispersion and information content of operator K. These are functions with exponential decay, used to scale the raw dispersion and information content of each individual. After a large number of initial tests (over a hundred different runs) we observed that $\mathcal{D}(I,X)$ and $\mathcal{I}_\mu$ varied within ranges of equal magnitude of approximately 6 units. In the case of $\mathcal{D}(I,X)$ the range was roughly $(9,15)$ and for $\mathcal{I}_\mu$ it was $(2.8,8.8)$. Therefore, the constants in f_2 and f_3 were set to $c_2 = 10$ and $c_3 = 3.8$, this gives approximately the same range of $(-1,5)$ for both functions; this is shown in Figure7(b).

Each objective function provide a non-linear scaling for the corresponding performance criteria used in this work. This does not affect the raw fitness ranking $R(i)$ of the SPEA2 algorithm. However, it does modify the density estimation $D(i)$ because it increases the penalization factor of Equation 8 in a non-linear manner. Therefore, the selection pressure pushes the population towards a highly dispersed Pareto front.



(a) (b)

Figure 7: Basic form of each objective function. (a) Shows the reciprocal function of f_1 in the range of $r_{K,J}(\epsilon)$. (b) Shows the exponential decay of functions f_2 and f_3 given parameters $c_2 = 10$ and $c_3 = 3.8$.

Figure 8: Samples from the van Gogh image sequence used for training by our MO-GP.

In summary, MO-GP searches for operators that are Pareto optimal based on the aforementioned objectives. These functions, however, do not make any assumptions regarding the underlying structure of the image region where an interest point is located, as stated in Section 2. In other words, in one sense the search is posed as an unsupervised learning problem since interest points are not predetermined within the training data, only the objective functions defined above are used to guide the search. This is contrary to the work of Ebner [9, 10] where a supervised approach is applied since the desired points are predetermined. Therefore, we should not expect that the evolved operators aim to select points that fulfill our own a priori expectations of what an interest point should be in terms of our definition of corner, blob or edge. Rather than that, the MO-GP will only minimize the above objective functions during the evolutionary search; hence, it optimizes the detection process based on well-established objective measures of performance.

5. Experimental results

Three goals are pursued with the experimental work:

1. The first goal is to study the relationship between the three performance criteria in order to

Table 1: GENERAL PARAMETER SETTINGS FOR THE MO/GP SEARCH.

Parameters	**Description and values**
Population	200 individuals.
Generations	50 generations.
Initialization	Ramped Half-and-Half.
Crossover & Mutation prob.	Crossover $p_c = 0.85$; Mutation $p_\mu = 0.15$.
Mating Selection	Binary Tournament.
Max. tree depth	3,5,7 & 9 levels.
Archive size	The SPEA2 archive size: 100.
Selection size	The number of individuals selected by SPEA2: 100.

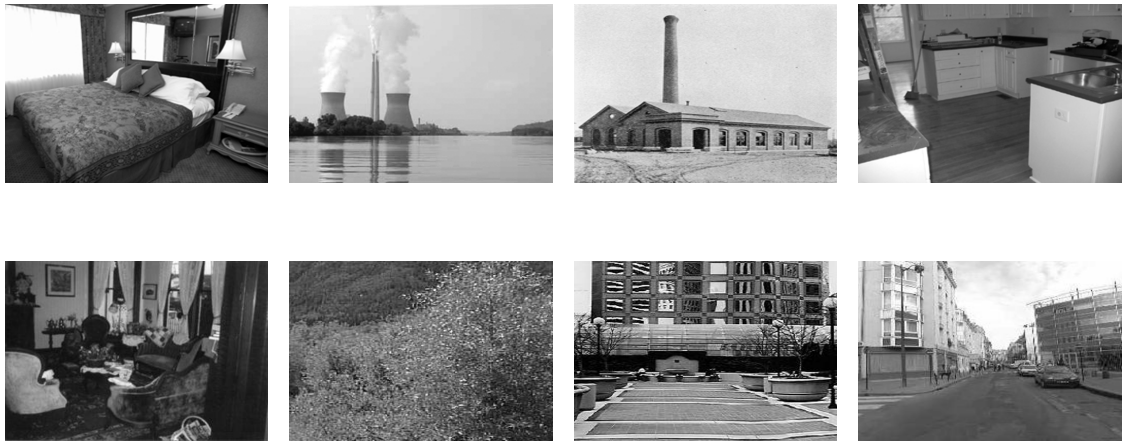


Figure 9: Sample images used to perform statistical validation of our evolved operators.

determine if a conflict exist between them. Therefore, the MO-GP is executed using every combination of two or more objectives: (1) Stability - Dispersion; (2) Stability - Information Content; (3) Dispersion - Information Content; and (4) Stability - Dispersion - Information Content.

2. Second, when a Pareto front is obtained, we compare the evolved operators with those previ-

ously proposed by human experts. Comparisons include the following man-made detectors: (1) Harris and Stephens (Harris) [38]; (2) Kitchen and Rosenfeld (K & R) [40]; (3) Beaudet [39]; and (4) Förstner [36]. For simplicity, we refer to these operators as Man-Made Interest Point operators, or *MMIP* operators. Additionally, the comparisons also include two operators that were evolved using the single objective approach (see Section 4.1), these are: K_{IPGP1} and K_{IPGP1} [14]. These are called Single Objective GP, or *SO-GP*, operators.

3. Finally, to validate the performance of the best evolved operators we include a test using as evaluation a large set of test images to confirm the performance trade-offs of each Pareto front.

5.1. Experimental Setup

The MO-GP was coded using the GPLAB Matlab toolbox [59], and a C++ implementation of the SPEA2 algorithm was downloaded from the Platform and Programming Language Independent Interface for Search Algorithms website[3]. The image sequence used for training was the van Gogh set of a planar scene with rotation transformations, all images are of size 348×512 pixels. The images were obtained from the website of the Lear team at INRIA Rhone- Alpes[4]. Matlab source code to compute the repeatability rate was downloaded from the website of the Visual Geometry Group at Oxford University[5]. The van Gogh sequence has one base image and sixteen progressively rotated images, each rotated 11.25° clockwise with respect to the previous one. Hence, the final image is rotated 180° with respect to the base image, Figure 8 shows some examples.

In previous work, experimental tests showed that the repeatability rate achieved by operators evolved with the van Gogh image, which is well-established in the literature, appears to provide a moderately dense sampling of data points; hence, the evolved operators attain high-fitness when they are tested on different images with different types of transformations [14, 12, 60]. In fact, previous results have shown that performance on this image sequence provides a good estimate of the stability of an interest point detector [8]. A plausible account for this outcome is that even a single image can contain a large and diverse set of possible interest point samples. Therefore, the learning process can compare the stability of many kinds of points and determine the optimal

[3] http://www.tik.ee.ethz.ch/sop/pisa/

[4] K. Mikolajczyk home page: http://lear.inrialpes.fr/people/mikolajczyk/ .

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

detection strategy given the performance objectives.

Additionally, a validation set of images is included to perform statistical comparisons between the evolved operators in order to confirm the dominance relations exhibited by Pareto Fronts produced by MO-GP. The goal is to provide statistical evidence that confirms that the trade-offs exhibited by the evolved operators are not artifacts of the training sequence. In other words, we validate the generality of the evolved operators by measuring their performance on a varied set of testing images. The validation set contains 500 images that were chosen randomly from the *scene categories* database that contains more than 3000 images from fifteen different types of scenes [61]. Figure 9 shows some of the images from the validation set which were scaled to a size of 512×348 pixels for simplicity. Moreover, in order to compare the performance of the operators we use the Kolmogorov-Smirnov non-parametric statistical test. Such statistical tests are used to compare the amount of point dispersion and information content of each operator; on the other hand, stability was not considered for further test mainly because the overall experimental evidence in the literature offers a favorable balance and also due to the complexity of calibrating multiple sequences of images.

Regarding the GP implementation, we use Koza's basic crossover and mutation operations [52], and all other parameters are summarized in Table 1. In each set of experiments, each with a different combination of objectives, the MO-GP is executed four times, once for every maximum allowed depth for the operator trees. Therefore, we present four Pareto front approximations for each combination of objectives. The sole exception is when all three objectives were considered simultaneously; in this case, a maximum depth of nine levels was used. From each set of results we present three representative solutions to illustrate the types of operators that the MO-GP generates. The mathematical expressions for each operator is slightly simplified in order to give a clear presentation of the results without significantly modifying the original expression generated by GP. Finally, it is important to note that the detection process for every operator K follows Eq. (1). However, instead of using a fixed threshold h, which could give a variable number of points depending on K, we choose the 500 points that have the highest response to K.

5.2. Stability vs Point Dispersion

The first set of experiments considers stability and point dispersion as optimization objectives. These objectives were also included in the single objective approach of [14], but a single evaluation

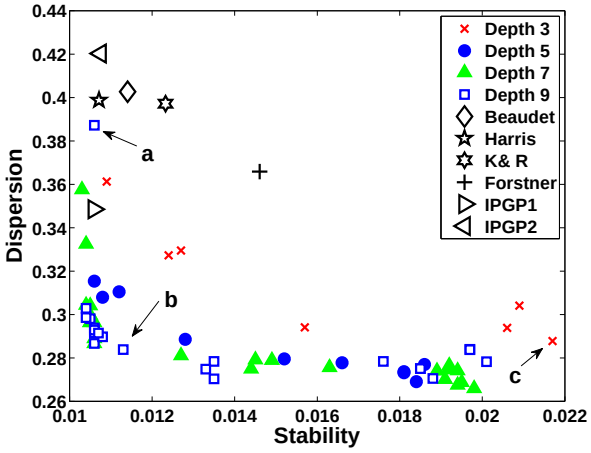


Figure 10: This graph shows the Pareto front of two objectives: stability and point dispersion, which was achieved by considering a maximum depth of four levels in the operator trees. The plot also shows the performance of four MMIPs (Harris, Beaudet, K & R, and Förstner), and two operators designed with the SO-GP approach, K_{IPGP1} and K_{IPGP2}. Finally, the plot also identifies three operators located at the extreme points of the Pareto front: (**a**), (**b**) and (**c**).

function fails to consider possible conflicts. For example, if interest points are clustered together with a low dispersion over the image plane they might still achieve a high repeatability rate if they are detected with an isotropic operator. Conversely, a detector that randomly selects points could conceivably achieve a high dispersion, but the detection would not be stable. In other words, the stability of a detector does not depend on the amount of point dispersion it produces. Therefore, these objectives should be addressed independently through a MO optimization.

Figure 10 presents the Pareto fronts generated by MO-GP using each of the maximum allowed depths. For comparative purposes, the figure also shows the performance of the four MMIPs (Harris, Beaudet, K & R and Förstner), and the two SO-GP operators (K_IPGP1 and K_{IPGP2}). It is important to mention that the implementation of Harris corresponds to the improved-Harris used by researchers in computer vision and their binaries can be obtained from INRIA and Oxford. In all of the runs, the MO-GP converges to a similar Pareto Front, except when the maximum allowed depth is set to three, in which case performance is inferior. In the other cases, the set of solutions found by the MO-GP dominate, in the Pareto sense, all of the other detectors included in the comparison. For instance, some of the MMIPs (Harris and Beaudet) achieve a comparable performance with respect to the stability criterion, but are definitely inferior with respect to point

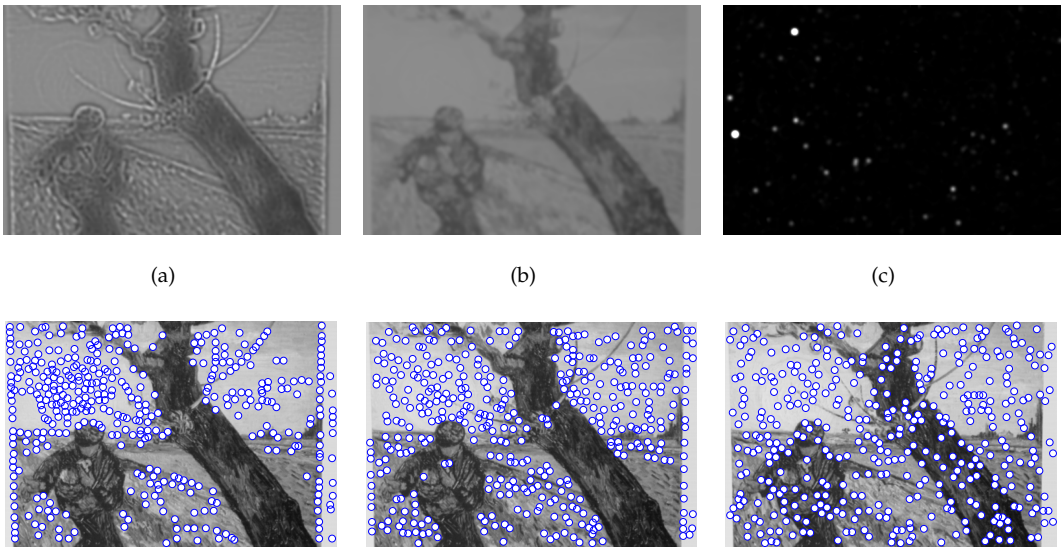


Figure 11: Interest image (first row) and interest points (second row) on the van Gogh image obtained with operators (**a**), (**b**) and (**c**) located on the Pareto front of Figure 10.

dispersion. Similarly, the two SO-GP operators also exhibit a high level of stability but poor point dispersion, this was expected given that they were evolved using the repeatability rate as the principal fitness component [14].

Figure 10 also identifies three operators located at what could be considered as extreme points of the Pareto Front: (**a**), (**b**) and (**c**). The differences between these operators can be qualitatively seen in Figure 11 where the interest image and the corresponding interest points are shown for the van Gogh image. The conflict between both objectives is evident, operator (**a**) detects stable points with a low dispersion, while operator (**c**) detects sparse and unstable points. The best compromise between both objectives is achieved by operator (**b**). The mathematical expressions for these operators are given in Table 2.

Figure 12 shows the types of interest points that each operators detects using two images from the validation set. The differences in dispersion are quite similar to that seen on the training image, with operator (**a**) achieving the lowest dispersion and (**c**) achieving the highest. Statistical comparisons regarding the amount of point dispersion show that the null-hypothesis, that both samples came from the same distribution, can be rejected between each of the operators at the 5% significance level. Figure 13 presents a box-plot comparison of the point dispersion obtained

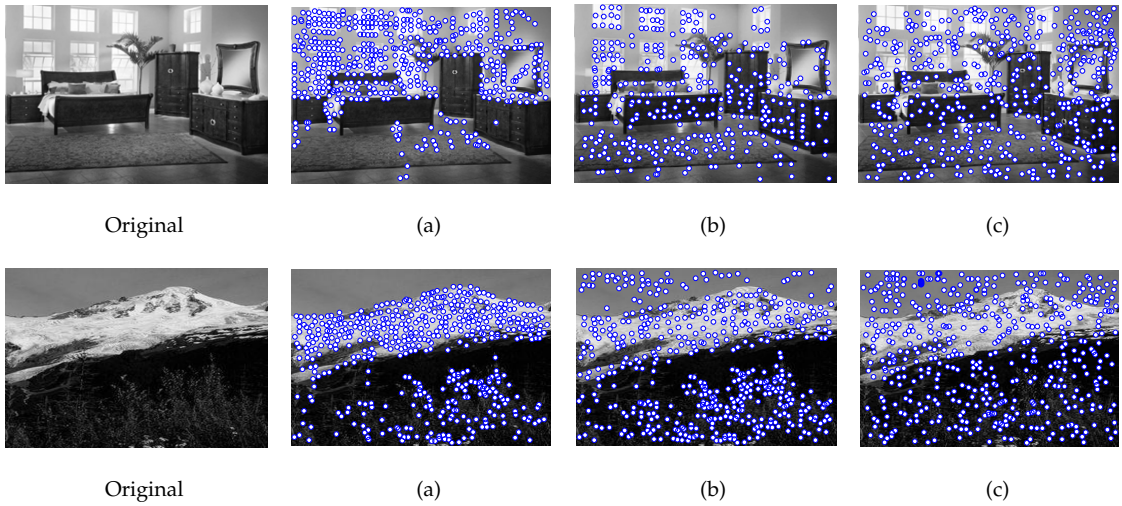


Figure 12: Interest points detected by operators (**a**), (**b**) and (**c**) on two images from the validation set.

with the entire validation set, showing the maximum, the minimum, the median and the 25th and 75th percentiles limits. These tests confirm that the dominance relations between the operators generalize quite well, with each operator achieving a significantly different amount of dispersion.

Let us now analyze these operators more closely. First, operator (**c**) is inversely proportional to the local curvature around each point computed along the vertical axis. Interest points are highly dispersed, but stability is poor because it only considers one principal direction. The same operation computed over the horizontal axis produces similar performance. On the other hand, operators (**a**) and (**b**) are in fact quite similar, both of them employ the absolute sum of three terms. First, a non-linear logarithmic term that depends on the image intensity at each point. The third term is the ratio between the weighted average intensity at each point computed with a Gaussian function and the intensity value. Hence, this term achieves a maximum for points that are darker than other points within its local Gaussian neighborhood. Indeed, the first and last terms in operators (**a**) and (**b**) are the same, the difference between them lies in the second term, a simple Difference-of-Gaussian (DoG) filter. Therefore, (**a**) and (**b**) can be seen as special cases of a more general operator. In this last case, we refine further the design as in previous man-made research and as a result we propose a novel interest point detector that the interested reader could find in the following article [15].

Table 2: SYMBOLIC EXPRESSIONS FOR OPERATORS (**A**), (**B**) AND (**C**) LOCATED ON THE PARETO FRONT OF FIGURE 10.

	Symbolic Expression
Operator (a):	$G_2 * \left\lvert G_1 * log\left(G_1 * I^2\right) + G_2 * (G_1 * I - I) + \frac{G_1 * I}{I} \right\rvert^2$
Operator (b):	$G_2 * \left\lvert G_1 * log(G_1 * I^2) + k \cdot G_2 * \lvert G_1 * I - I \rvert + \frac{G_1 * I}{I} \right\rvert^2$
Operator (c):	$G_2 * \left(\frac{L_y}{L_{yy}}\right)$

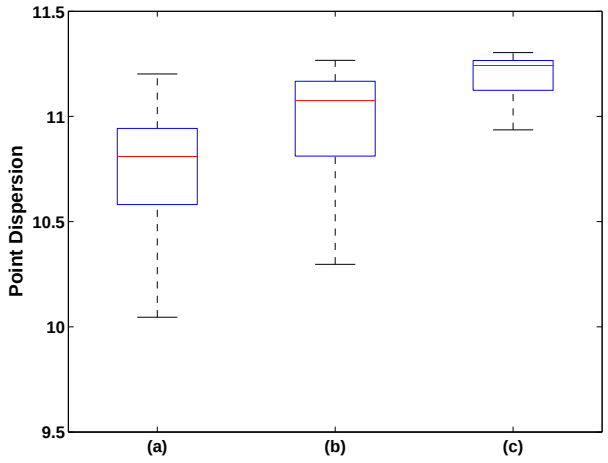
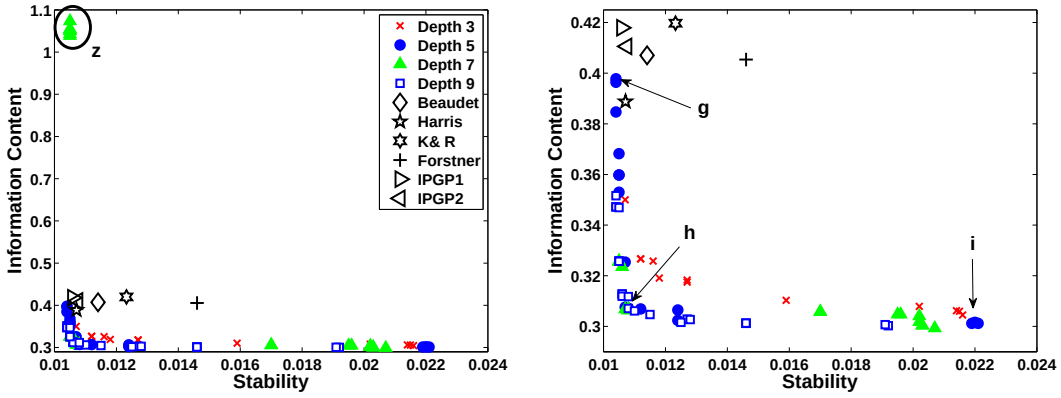


Figure 13: Box-plot comparison for the amount of point dispersion obtained by operators (**a**), (**b**) and (**c**) on the validation set.

5.3. Stability vs Information Content

Here again, it is reasonable to assume that these criteria are also in conflict for the following reasons. Firstly, points that are very close together will undoubtedly share a similar local neighborhood, and will thus share a similar description. Secondly, according to the results from previous experiments, the cluttered points that are computed within a region can be detected in an stable way. Thirdly, some operators detected points that lie over a region with a similar structure, such as lines, edges or blobs, and these points will tend to be stable. Thus, because they lie on simi-



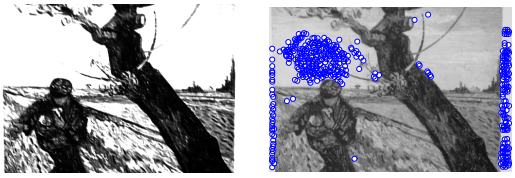
(a) (b)

Figure 14: These graphs show Pareto front between two objectives: stability and information content, which were achieved by considering a maximum depth of four levels in the operator trees. (a) Comparison with MMIP and SO-GP operators, and the extreme operator (**z**). (b) A closer look at the Pareto front with three operators located at possible extreme points: (**g**), (**h**) and (**i**).

lar structures their corresponding descriptive vectors will also be similar. Therefore, we conclude that the evolution of interest operators based on these criteria should be carried out using a MO approach.

The Pareto fronts plotted in Figure 14 show how the MMIP and SO-GP operators are mostly dominated by Pareto optimal solutions found by the MO-GP. The MMIP and SO-GP operators achieve a stable detection but tend to provide less information content. These results illustrate the fact that the information content criterion was not explicitly considered during the design of the MMIP or SO-GP operators. In Figure 14a, three operators that achieve an extreme performance are highlighted; it demonstrates a high stability and low information content. To illustrate this result, Figure 15 shows the interest image and interest points detected with one of these operators, it is named operator (**z**) which is given by $\left(L_{xy} + 2I + k\left(L_{xy} + I + kL_{xx}\right)^2\right)^2$. The figure shows that interest points detected with (**z**) exhibit a very low dispersion and information content. This suggest that interest point dispersion and information content are indeed related, however this idea is further explored in the following section.

Figure 14b identifies three operators on the Pareto Front: (**g**), (**h**) and (**i**). Their corresponding interest image and detected interest points on the training image are shown in Figure 16, and their



(a) (b)

Figure 15: Interest point detection with operator (**z**), located on the extreme end of the Pareto front shown in Figure 14a.

mathematical expressions are given in Table 3. The amount of information content extracted by each operator was compared using the validation set and in all three cases the null-hypothesis was rejected at the 5% significance level, see Figure 17.

Operator (**g**) is located near the extreme of the Pareto Front of Figure 14b; it achieves a stable detection but conveys a small amount of information. The performance of this operator is similar to that achieved by the MMIP and SO-GP operators. The symbolic expression of operator (**g**) reveals that it basically consists of a DoG filter, thus most of the points it detects lie on edges or borders. For this reason, the descriptors extracted from these points tend to be similar. This result also explains the performance of the Harris detector, for example, which mostly detects image corners, features that also share a similar structure. At the other end of the Pareto front of Figure 14b it is located operator (**i**), which can be seen as the opposite of operator (**g**); in other words, it detects very diverse points with a high information content but low stability. The best trade-off between these objectives is obtained by operator (**h**), which has a very unorthodox mathematical expression, unlike any of the operators found in modern literature. However, the combination of stable detection and a high information content makes operator (**h**) a promising new tool for vision researchers that are interested in designing object recognition systems based on salient and distinctive features [1, 5]. A qualitative comparison of operators (**g**), (**h**) and (**i**) is shown in Figure 18 using two images from the validation set.

5.4. Point Dispersion vs Information Content

The final pair of objectives is point dispersion and information content, which appears to be positively correlated. For instance, if interest points are highly dispersed, then it is reasonable to assume that their corresponding descriptors will be different, thus a high information content could be expected. Conversely, if interest points lie very close together within the image, then

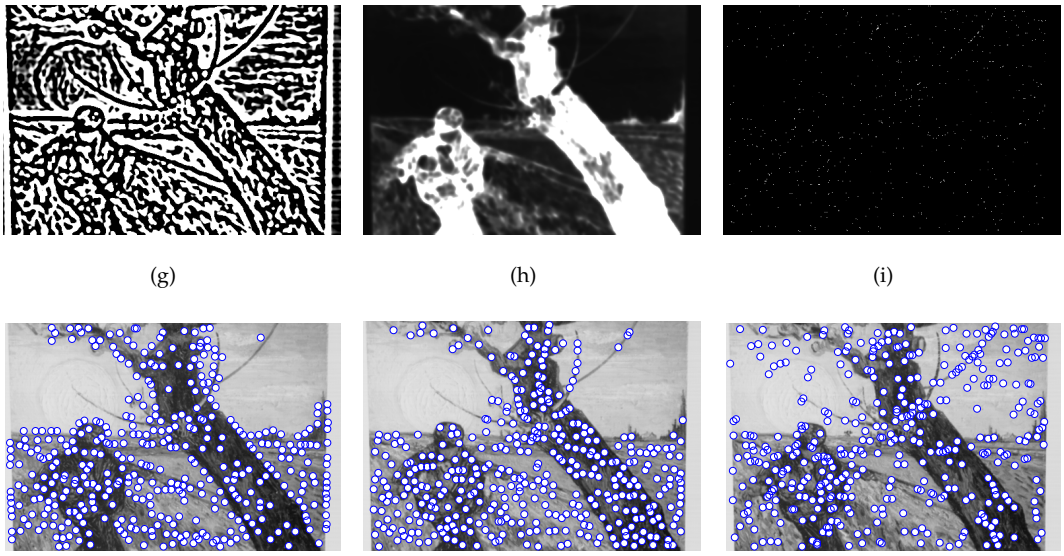


Figure 16: Interest image (first row) and interest points (second row) on the van Gogh image obtained with operators (**g**), (**h**) and (**i**) located on the Pareto front of Figure 14.

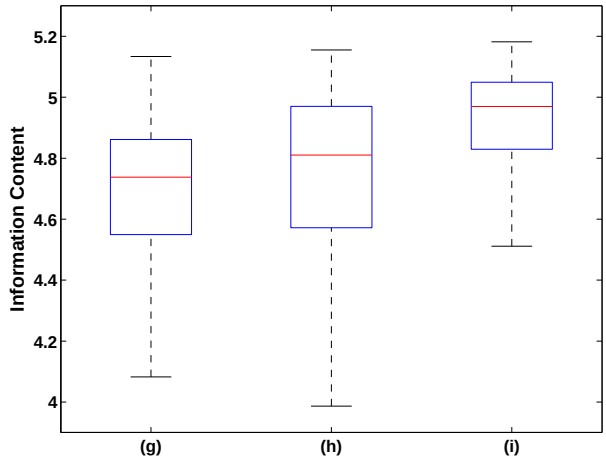


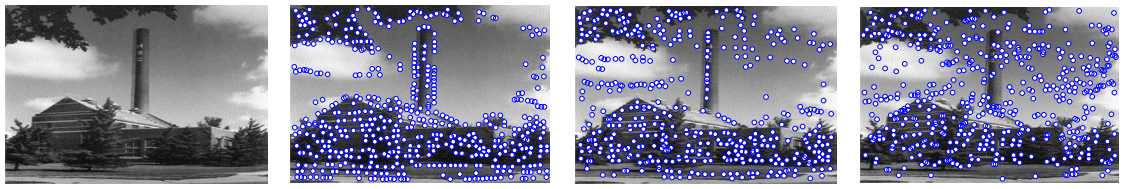
Figure 17: Box-plot comparison for the amount of information content obtained by operators (**g**), (**h**) and (**i**) on the validation set.

we can assume that their descriptors will be very similar, and they will thus convey a low information content. Experimental results, however, suggest that these assumptions are not without exceptions.

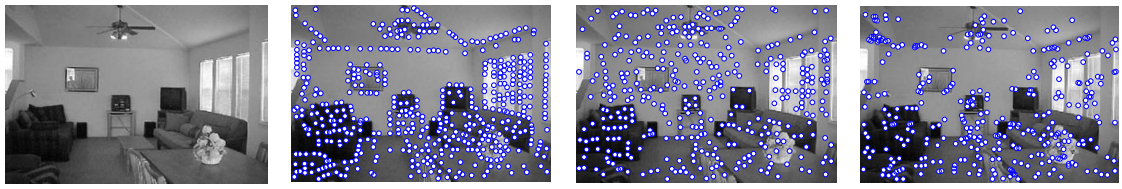
The Pareto fronts for these experiments are shown in Figure 19. The plot reveals that the MMIP

Table 3: SYMBOLIC EXPRESSIONS FOR OPERATORS (**G**), (**H**) AND (**I**) LOCATED ON THE PARETO FRONT OF FIGURE 14.

	Symbolic Expression
Operator (g):	$G_2 * G_1 * (I - G_2 * I)$
Operator (h):	$\frac{\sqrt{G_1 * \lvert G_2 * I \rvert}}{\lvert G_2 * ((G_2 * I)^2 - \lvert\lvert L_{xy} + I \rvert + k \cdot L_{xx} \rvert)\rvert}$
Operator (i):	$G_1 * (k \cdot L_{yy}) - \left\lvert \lvert L_{yy} - L_{xy} \rvert - \frac{log(I)}{L_y} \right\rvert$



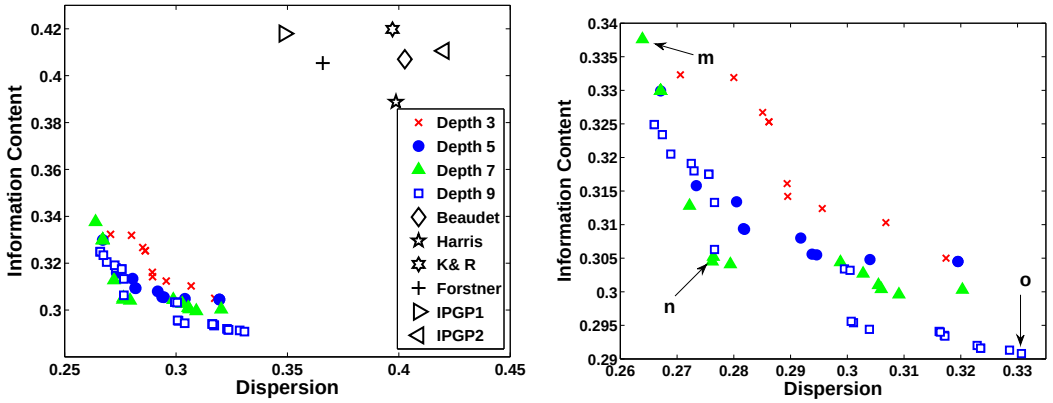
Original (g) (h) (i)



Original (g) (h) (i)

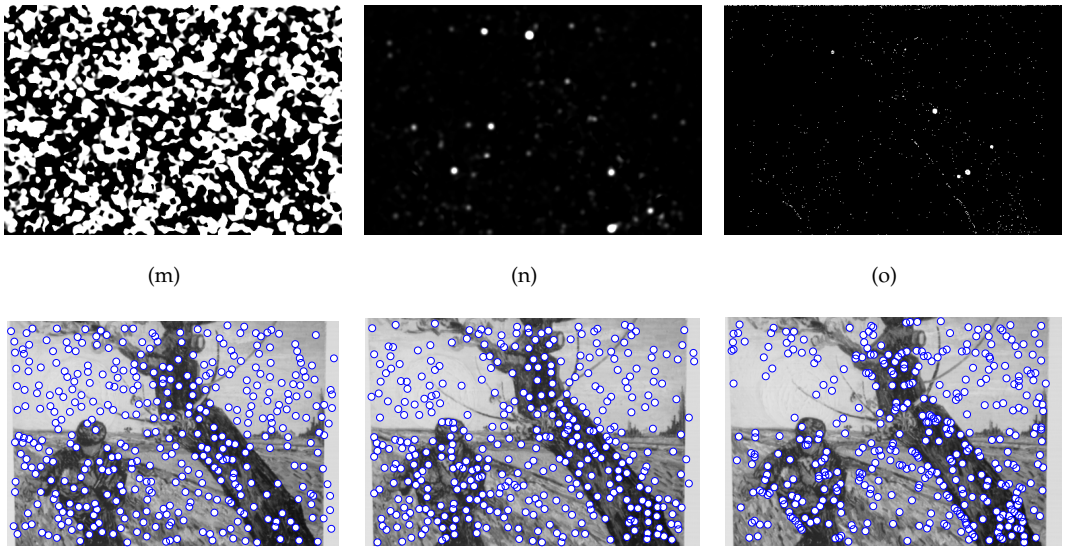
Figure 18: Interest points detected by operators (**g**), (**h**) and (**i**) on two images from the validation set.

and SO-GP operators are dominated by the Pareto optimal solutions; this was expected given the results of the previous sections. Figure 19b identifies three operators on the Pareto front: (**m**), (**n**) and (**o**). The corresponding interest image and interest points on the training image are shown in Figure 20, and Table 4 gives the mathematical expressions for each one. These operators are compared according to the information content and point dispersion achieved on the validation set. Thus, according to dispersion the difference in performance between each operator is statistically significant at the 5% level. The results on the validation set is illustrated in the box-plot of Figure



(a) (b)

Figure 19: The Pareto front found between the point dispersion and information content using a maximum depth of four levels in the operator trees. (a) Comparison with MMIP and SO-GP operators. (b) A closer look at the Pareto front with three operators located at extreme points: (**m**), (**n**) and (**o**).



(m) (n) (o)

Figure 20: Interest image (first row) and interest points (second row) on the van Gogh image obtained with operators (**m**), (**n**) and (**o**) located on the Pareto front of Figure 19.

21. However, with respect to information content the results are slightly different. On the one hand, (**m**) and (**n**) are both statistically different from (**o**) at the 5% significance level. On the other hand, the null-hypothesis cannot be rejected between operators (**m**) and (**n**), see Figure 22.

Table 4: SYMBOLIC EXPRESSIONS FOR OPERATORS (**M**), (**N**) AND (**O**) LOCATED ON THE PARETO FRONT OF FIGURE 19.

	Symbolic Expression
Operator (m):	$G_2 * G_2 \left(\frac{L_y}{2 \cdot L_{yy} + L_{xy}} \right)$
Operator (n):	$\frac{G_2 * G_2 * \left(\frac{L_{xy}}{2 \cdot L_{yy} +_L xy + L_x} \right)}{G_1 * G_2 * G_1 * G_2 * G_1 * I}$
Operator (o):	$\frac{G_2 * \left[G_2 * G_1 * I + G_2 * \left(\frac{L_{xy}}{G_2 * L_y} \right) \right]}{log(L_y) - \left\| \sqrt{I} - \lvert L_{xx} + L_{xy} \rvert + G_1 * L_x \right\| - \left\| G_1 * G_2 * \lvert I - L_y \rvert - \left\| \lvert L_y + k \cdot I \rvert + \frac{I + Lxy}{Lyy \cdot Lxx} \right\| \right\|}$

These results appear to be counterintuitive. They seem to suggest that even if points are highly dispersed their descriptors might still be similar and vice versa. The points extracted with (**m**) convey a lower amount of information content than the points detected by (**o**), even if the amount of point dispersion is significantly higher for (**m**) with respect to (**o**). However, this result is not completely unexpected. For instance, in [8] the authors report that the Harris detector extracts higher information content than a set of random points. Thus, it could be argued that the set of points detected by (**m**) are similar to a random selection process. Similarly, operator (**o**) extracts points with a lower amount of dispersion, but the image structures it selects provide a more varied set of local descriptors. Therefore, it seems that even though information content and dispersion appear to be related, this relationship will depend upon the underlying structures that each operator detects. Finally, Figure 23 shows a qualitative comparison using images from the validation set.

5.5. Stability, Point Dispersion, and Information Content

The final experiment considers all three objectives simultaneously. However, the results presented above suggest that the best approximation of the true Pareto fronts were obtained when the maximum tree depth was set to seven or nine levels. Hence, the algorithm was tested with a maximum allowed depth of nine levels, and the results are shown in Figure 24 using two different views. Similarly to what was obtained in the other experimental tests, we can observe that the MO-GP algorithm finds a Pareto surface of non-dominated solutions. Experimental tests on these operators show similar trends to those described previously, with optimal operators located in

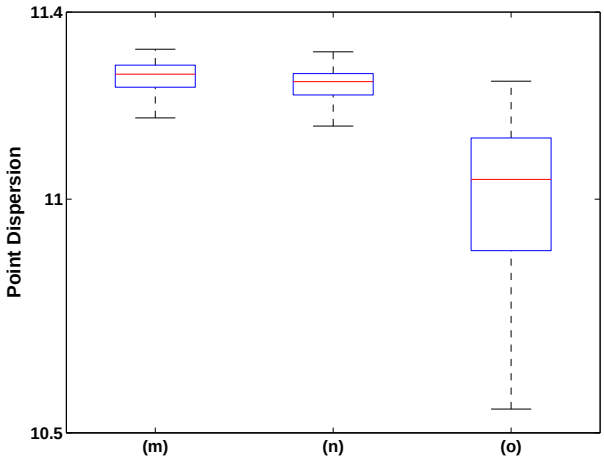


Figure 21: Box-plot comparison for the amount of point dispersion obtained by operators (**m**), (**n**) and (**o**) on the validation set.

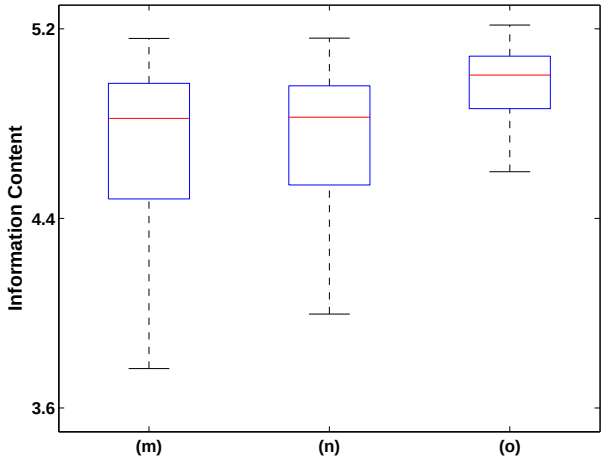


Figure 22: Box-plot comparison for the amount of information content obtained by operators (**m**), (**n**) and (**o**) on the validation set.

different points of the Pareto Front, in this case a surface, exhibiting different trade-offs among the objectives. However, for the sake of brevity we omit any further details regarding these operators.

5.6. Computational Cost

Finally, let us consider the overall complexity and computational cost of the proposed algorithm. It must be stated that as expected from previous work [14, 54], the computational cost of

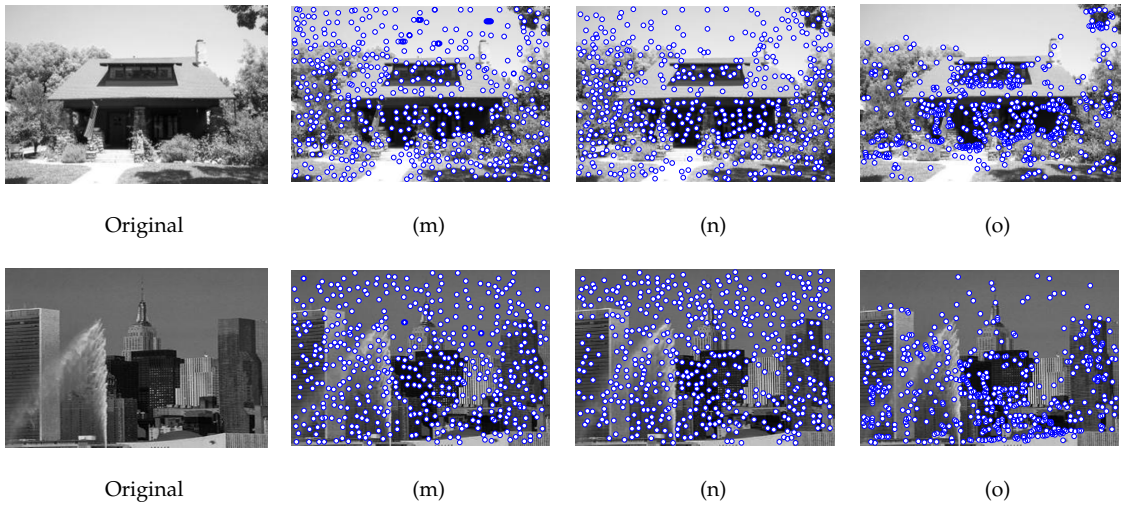
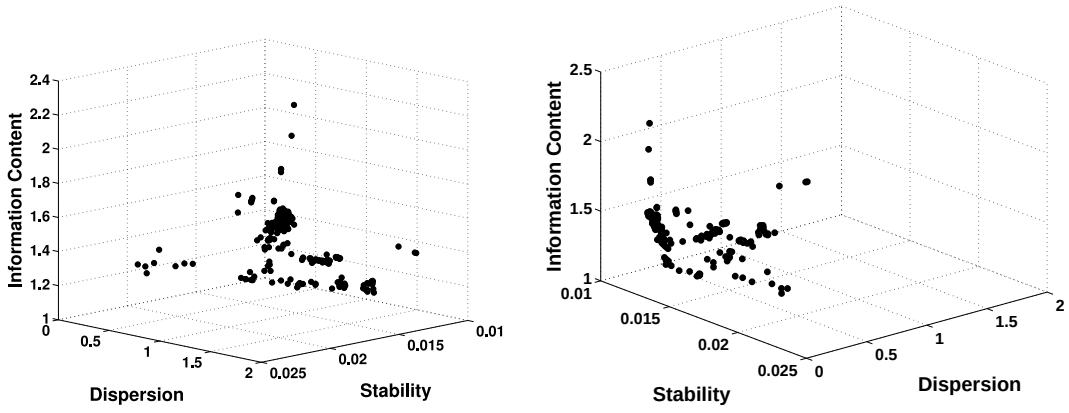


Figure 23: Interest points detected by operators (**m**), (**n**) and (**o**) on two images from the validation set.



(a) (b)

Figure 24: Two views of the Pareto front obtained by the MO-GP using the three objectives simultaneously: stability, point dispersion and information content.

the MO-GP search is quite substantial. In fact, each run of the algorithm required between two and four days of computation on a regular desktop PC running Linux and Matlab 2007, such amount of computational time should not be taken lightly. However, once the learning process has been completed, the MO-GP does not need to be executed again, especially since the experimental validation of the evolved operators showed that their performance was consistent with that achieved during testing. Therefore, once the MO-GP search has finished, we are now free to

Table 5: Average computational time, in seconds, for each of the evolved operators computed on the validation set.

Operator	**(a)**	**(b)**	**(c)**	**(g)**	**(h)**	**(i)**	**(m)**	**(n)**	**(o)**
Time (sec.)	0.05	0.05	0.03	0.02	0.06	0.06	0.06	0.12	0.19

use the evolved interest operators on-line. In this case, the performance of the operators is quite reasonable. For example, if we compute the interest operator for all images on the validation test, 500 in total, the average run time of each operator is presented in Table 5. Considering that these are the complete program trees, without simplification and using non-optimal Matlab code, the average computational time is good.

Moreover, some of the more efficient operators, such as **(a)**, **(b)**, **(g)** and **(m)**, were ported to a C++ implementation using the vision library LibCVD[6], which allowed us to obtain a frame-rate of 30 fps[7] with a video resolution of 640×480 on a Unibrain Fire-i FireWire camera.

6. Summary and conclusions

In this paper, we present an approach based on mutiobjective genetic programming for the automatic synthesis of image operators that detect interesting image points. Interest point detectors are an important tool for many modern computer vision systems that perform tasks such as scene recognition, object detection and image indexing. However, no previous proposals have explicitly considered the multiobjective nature of this task. In the present work, a MO-GP is used to produce a variety of interest point operators that provide different trade-offs among the most relevant performance criteria from computer vision literature. The evolved operators provide new detection strategies that are significantly different from previous proposals. For instance, some operators provide a very stable detection process that might be useful for object tracking, while others focus on detecting highly informative image points which can be used for recognition or detection tasks.

In sum, the Pareto fronts found by the GP search illustrate the intrinsic multiobjective nature of the point detection task and suggest that a system designer must consider the conflicting nature between the performance objectives. Moreover, the evolved operators substantially outperform

[6] http://www.edwardrosten.com/cvd/index.html

[7] Note that this is the maximum frame rate of the Fire-i camera, but the program could process information faster if appropriate hardware is provided.

several man-made detectors and also dominate previous operators evolved through a single objective approach. These claims were all validated using statistical tests on a wide variety of images and illustrated qualitatively on several real-world images.

Bibliography

[1] C. Schmid, R. Mohr, Local grayvalue invariants for image retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 19 (5) (1997) 530–534.

[2] D. G. Lowe, Object recognition from local scale-invariant features., in: Proceedings of the International Conference on Computer Vision (ICCV), 20-25 September, Kerkyra, Corfu, Greece, Vol. 2, IEEE Computer Society, 1999, pp. 1150–1157.

[3] T. Tuytelaars, K. Mikolajczyk, Local invariant feature detectors: a survey, Found. Trends Comput. Graph. Vis. 3 (3) (2008) 177–280.

[4] Q. Li, J. Ye, C. Kambhamettu, Interest point detection using imbalance oriented selection, Pattern Recogn. 41 (2) (2008) 672–688.

[5] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, IEEE Trans. Pattern Anal. Mach. Intell. 27 (10) (2005) 1615–1630.

[6] L. Trujillo, G. Olague, P. Legrand, E. Lutton, Regularity based descriptor computed from local image oscillations, Optics Express 15 (10) (2007) 6140–6145.

[7] M. Heikkil, M. Pietikinen, C. Schmid, Description of interest regions with local binary patterns, Pattern Recogn. 42 (3) (2009) 425 – 436.

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

[9] M. Ebner, On the evolution of interest operators using genetic programming, in: R. Poli, et al. (Eds.), Late Breaking Papers at EuroGP'98, The University of Birmingham, UK, 1998, pp. 6–10.

[10] M. Ebner, A. Zell, Evolving task specific image operator., in: R. Poli, et al. (Eds.), First European Workshops, EvoIASP'99 and EuroEcTel'99, Göteborg, Sweden, May 26-27, Vol. 1596 of LNCS, 1999, pp. 74–89.

[11] L. Trujillo, G. Olague, Synthesis of interest point detectors through genetic programming, in: M. Cattolico (Ed.), Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), Seattle, Washington, July 8-12, Vol. 1, ACM, 2006, pp. 887–894.

[12] L. Trujillo, G. Olague, Scale invariance for evolved interest operators, in: Proceedings of EvoIASP2007 Ninth European Workshop on Evolutionary Computation in Image Analysis and Signal Processing, incorporated in Evo* 2007, 11-13 April, Valencia, Spain, LNCS, Springer-Verlag, 2007, pp. 423–430.

[13] L. Trujillo, G. Olague, E. Lutton, F. Fernández de Vega, Multiobjective design of operators that detect points of interest in images, in: M. Cattolico (Ed.), Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), Atlanta, GA, July 12-16, ACM, New York, NY, USA, 2008, pp. 1299–1306.

[14] L. Trujillo, G. Olague, Automated design of image operators that detect interest points, Evol. Comput. 16 (4) (2008) 483–507.

[15] G. Olague, L. Trujillo, Evolutionary-computer-assisted design of image operators that detect interest points using genetic programming, Image Vision Comput. 29 (7) (2011) 484–498.

[16] C. B. Pérez, G. Olague, Learning invariant region descriptor operators with genetic programming and the f-measure, in: 19th International Conference on Pattern Recognition (ICPR 2008), December 8-11, 2008, Tampa, Florida, USA, IEEE, 2008, pp. 1–4.

[17] C. B. Perez, G. Olague, Evolutionary learning of local descriptor operators for object recognition, in: GECCO '09: Proceedings of the 11th Annual conference on Genetic and evolutionary computation, ACM, New York, NY, USA, 2009, pp. 1051–1058.

[18] S. Cagnoni, E. Lutton, G. Olague (Eds.), Genetic and Evolutionary Computation for Image Processing and Analysis, Vol. 8 of EURASIP Book Series on Signal Processing and Communications, Hindawi Publishing Corporation, 2007.

[19] K. Krawiec, Genetic programming-based construction of features for machine learning and knowledge discovery tasks, Genetic Programming and Evolvable Machines 3 (4) (2002) 329–343.

[20] K. Krawiec, D. Howard, M. Zhang, Overview of object detection and image analysis by means of genetic programming techniques, in: Frontiers in the Convergence of Bioscience and Information Technologies 2007, FBIT 2007, Jeju Island, Korea, October 11-13, IEEE Computer Society, 2007, pp. 779–784.

[21] Y. Zhang, P. I. Rockett, Evolving optimal feature extraction using multi-objective genetic programming: a methodology and preliminary study on edge detection, in: GECCO '05: Proceedings of the 2005 conference on Genetic and evolutionary computation, ACM, New York, NY, USA, 2005, pp. 795–802.

[22] L. Trujillo, P. Legrand, J. Lévy-Véhel, The estimation of hölderian regularity using genetic programming, in: GECCO '10: Proceedings of the 12th annual conference on Genetic and evolutionary computation, ACM, New York, NY, USA, 2010, pp. 861–868.

[23] B. Bhanu, Y. Lin, Object detection in multi-modal images using genetic programming, Appl. Soft Comput. 4 (2) (2004) 175 – 201.

[24] K. Krawiec, B. Bhanu, Visual learning by coevolutionary feature synthesis, IEEE Trans. Syst., Man, Cybern, Part B 35 (3) (2005) 409–425.

[25] R. Poli, Genetic programming for feature detection and image segmentation, in: T. C. Forgarty (Ed.), AISB Workshop Evolutionary Computing, 1996, pp. 110–125.

[26] D. Howard, S. C. Roberts, C. Ryan, Pragmatic genetic programming strategy for the problem of vehicle detection in airborne reconnaissance, Pattern Recogn. Lett. 27 (11) (2006) 1275–1288.

[27] M. Zhang, V. B. Ciesielski, P. Andreae, A domain-independent window approach to multiclass object detection using genetic programming, EURASIP Journal on Applied Signal Processing, Special Issue on Genetic and Evolutionary Computation for Signal Processing and Image Analysis 2003 (8) (2003) 841–859.

[28] A. Song, V. Ciesielski, Texture segmentation by genetic programming, Evol. Comput. 16 (4) (2008) 461–481.

[29] R. da S. Torres, A. X. Falco, M. A. Gonalves, J. P. Papa, B. Zhang, W. Fan, E. A. Fox, A genetic programming framework for content-based image retrieval, Pattern Recogn. 42 (2) (2009) 283 – 292.

[30] E. Dunn, G. Olague, Multi-objective sensor planning for efficient and accurate object reconstruction, in: Applications of Evolutionary Computing, EvoWorkshops 2004: EvoBIO, EvoCOMNET, EvoHOT, EvoIASP, EvoMUSART, and EvoSTOC, Coimbra, Portugal, April 5-7, 2004, Proceedings, Vol. 3005 of Lecture Notes in Computer Science, Springer, 2004, pp. 312–321.

[31] K. Krawiec, Generative learning of visual concepts using multiobjective genetic programming, Pattern Recogn. Lett. 28 (16) (2007) 2385–2400.

[32] Y. Zhang, P. I. Rockett, A generic multi-dimensional feature extraction method using multiobjective genetic programming, Evol. Comput. 17 (1) (2009) 89–115.

[33] J. McGlone, et al. (Eds.), Manual of photogrammetry, American Society of Photogrammetry and Remote Sensing, 2004.

[34] G. Olague, B. Hernández, A new accurate and flexible model based multi-corner detector for measurement and recognition, Pattern Recogn. Lett. 26 (1) (2005) 27–41.

[35] C. S. Kenney, M. Zuliani, B. S. Manjunath, An axiomatic approach to corner detection, in: International Conference on Computer Vision and Pattern Recognition (CVPR), 2005, pp. 191–197.

[36] W. Förstner, E. Gülch, A fast operator for detection and precise location of distinct points, corners and centres of circular features, in: ISPRS Intercommission Conference on fast processing of photogrammetric data, 1987, pp. 149–155.

[37] J. Shi, C. Tomasi, Good features to track, in: Proceedings of the 1994 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94), June 1994, Seattle, WA, USA, IEEE Computer Society, 1994, pp. 593–600.

[38] C. Harris, M. Stephens, A combined corner and edge detector, in: Proceedings from the Fourth Alvey Vision Conference, Vol. 15, 1988, pp. 147–151.

[39] P. R. Beaudet, Rotational invariant image operators, in: Proceedings of the 4th International Joint Conference on Pattern Recognition (ICPR),Tokyo, Japan, 1978, pp. 579–583.

[40] L. Kitchen, A. Rosenfeld, Gray-level corner detection, Pattern Recogn. Lett. 1 (1982) 95–102.

[41] A. Noble, Descriptions of image surfaces, Ph.D. thesis, Department of Engineering Science, Oxford University (1989).

[42] P. Tissainayagam, D. Suter, Assessing the performance of corner detectors for point feature tracking applications., Image Vision Comput. 22 (8) (2004) 663–679.

[43] F. Mokhtarian, F. Mohanna, Performance evaluation of corner detectors using consistency and accuracy measures, Comput. Vis. Image Underst. 102 (1) (2006) 81–94.

[44] A. J. Davison, I. D. Reid, N. D. Molton, O. Stasse, Monoslam: Real-time single camera SLAM, IEEE Trans. Pattern Anal. Mach. Intell. 29 (6) (2007) 1052–1067.

[45] G. Yang, C. V. Stewart, M. Sofka, C.-L. Tsai, Registration of challenging image pairs: Initialization, estimation, and decision, IEEE Trans. Pattern Anal. Mach. Intell. 29 (11) (2007) 1973–1989.

[46] J. J. Koenderink, A. J. van Doom, Representation of local geometry in the visual system, Biol. Cybern. 55 (6) (1987) 367–375.

[47] D. G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vision 60 (2) (2004) 91–110.

[48] L. Trujillo, P. Legrand, G. Olague, C. Pérez, Optimization of the hölder image descriptor using a genetic algorithm, in: GECCO '10: Proceedings of the 12th annual conference on Genetic and evolutionary computation, ACM, New York, NY, USA, 2010, pp. 1147–1154.

[49] S. Mallat, A wavelet tour of signal processing, 2nd Edition, Elsevier, San Diego, CA, 1999.

[50] C. Tricot, Curves and Fractal Dimension, Springer-Verlag, 1995.

[51] K. De Jong, Evolutionary Computation: A Unified Approach, The MIT Press, 2001.

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

[53] H. P. Moravec, Towards automatic visual obstacle avoidance, in: IJCAI, 1977, p. 584.

[54] D. Lombraña, F. Fernández de Vega, G. Olague, L. Trujillo, B. Segal, Customizable execution environments with virtual desktop grid computing, in: PDCS 2007: Proceedings of the 19th IASTED International Conference on Parallel and Distributed Computing Systems, 19-21 November, Cambridge MA, USA, Morgan Kaufmann Publishers Inc., 2007, pp. 7–12.

[55] V. Pareto, Cours D'Economie Politique, Rouge, Lausanne, 1896.

[56] C. Coello, D. V. Veldhuizen, G. Lamont, Evolutionary Algorithms for Solving Multi-Objective Problems, Kluwer Academic Publishers, New York, New York, 2002.

[57] E. Zitzler, M. Laumanns, S. Bleuler, A tutorial on evolutionary multiobjective optimization, in: X. Gandibleux, et al. (Eds.), Metaheuristics for Multiobjective Optimisation, Lecture Notes in Economics and Mathematical Systems, Springer-Verlag, 2004, pp. 3–38.

[58] E. Zitzler, M. Laumanns, L. Thiele, Spea2: Improving the strength pareto evolutionary algorithm for multiobjective optimization, in: Evolutionary Methods for Design, Optimisation, and Control, 2002, pp. 19–26.

[59] S. Silva, J. Almeida, Gplab–a genetic programming toolbox for matlab, in: L. Gregersen (Ed.), Proceedings of the Nordic MATLAB conference, 2003, pp. 273–278.

[60] G. Olague, L. Trujillo, A Genetic Programming Approach to the Design of Interest Point Operators, Vol. 256 of Studies in Computational Intelligence, Springer-Verlag, 2009, Ch. 5, pp. 81–91.

[61] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories., in: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 17-22 June, New York, NY, Vol. 2, IEEE Computer Society, 2006, pp. 2169–2178.