Lakemond, Ruan and McKinnon, David N. R. and Fookes, Clinton and Sridharan, Sridha (2007) A Feature Clustering Algorithm for Scale-space Analysis of Image Structures. In *Proceedings International Conference on Signal Processing and Communication Systems 2007*, Gold Coast, Australia.

A Feature Clustering Algorithm for Scale-space Analysis of Image Structures

Ruan Lakemond, David N. R. McKinnon, Clinton Fookes and Sridha Sridharan
Image and Video Research Laboratory
Queensland University of Technology
GPO Box 2434, 2 George St
Brisbane, Queensland 4001
{*r.lakemond*, *d.mckinnon*, *c.fookes*, *s.sridharan*}@qut.edu.au

***Abstract*—In describing image features it is important to consider the fact that the appearance of a feature depends on the scale or resolution at which it is observed. Existing robust image feature detectors address the issue by selecting a characteristic scale for each detected feature and subsequently describing the feature as it appears at its characteristic scale. A new method is presented for the multi-scale analysis of derivative based image features that represents a 2D image feature by its locus in scale-space. An algorithm is also presented for efficiently producing the discrete loci representations of image features through clustering features detected at multiple scales. This new method provides an entry point to potential multi-scale descriptions of image features, as well as new possibilities for feature set reduction and filtering.**

I. Introduction

In attempting to describe 2D image features for the purpose of analyzing or describing the 3D objects of which the image is a projection, it is important to account for the scaling and deformation that result from 3D to 2D projection. Robust detectors aim to locate features in a repeatable manner despite changes in lighting, rotation, scale and projective deformation. Local features and descriptors provide a compact means for describing complex scenes, while partially avoiding complex global transformations. These detectors are typically applied to applications such as wide base-line matching and calibration, as well as object recognition and classification.

Derivative based feature detectors such as the Difference of Gaussian detector [1], [2], Determinant of Hessian detector [3] and Harris detector [4] operate by producing a saliency map of an image. The saliency map is a function of image derivative and indicates regions of high information content. Features are located by detecting local maxima in the saliency map. This class of detectors is in general sensitive to the scale or resolution of the image. A multi-scale analysis is required to achieve robustness to scale change.

A scale-space representation of an image is obtained by filtering the original image with Gaussian kernels of different standard deviations. The scale space then has two spatial dimensions and a scale dimension corresponding to the standard deviation of the applied Gaussian kernel. Discrete scale-space representations are often referred to as scale-space pyramids. Lindeberg presents an extensive study on scale-space in [5] and a study specifically on gradient based image features in [6].

Several attempts have been made to address the problem of scale sensitivity in image feature detection. There are several difficulties facing the existing detectors, as will be discussed in detail in the next section. These include excessive features, the need to select the good features from amongst the poor features, and computational cost. This paper explores a new method for the scale-space analysis of image features that addresses and improves upon the listed difficulties.

Section II presents existing scale-space feature analysis techniques. Section III presents the new feature locus representation of features, as well as an efficient method for producing this representation for practical features. Analysis of applications using the locus representation are discussed in Section IV. Scale selection is demonstrated as a simple application of the new locus representation in Section IV-A and its benefits over simple scale-space maxima are demonstrated.

II. Scale-space Analysis of Features

The output of various image feature detectors is known to be sensitive to a change in scale. In order to achieve scale invariance, features need to be examined over a range of scales and described using a scale invariant method. One option is to simply detect features at multiple scales, as in [7]. The result is a very large and dense set of features, most of which are not likely to yield appropriate matches when matching against a set of features from a different viewpoint. It may be undesirable to have excessively large numbers of features for short and medium base-line problems. In general large sets of features that yield few correspondences present a problem in terms of computational efficiency. Adaptation processes applied to features in order to achieve robustness to affine deformations (for example [8]) cause features to converge to stable points in affine scale-space. Applying such an adaptation process to an excessively large set of features results in a large amount of redundant processing, as the features will converge to a much smaller set of points. A smaller set of stable features is desired as a starting point for adaptation.

The most common approach to addressing the scale invariance problem is to select a feature at a characteristic scale and describe the feature as it appears at the selected scale. A characteristic scale is selected for a feature by selecting a scale where an interesting and repeatable occurrence is found.

The simplest method of scale selection is to locate scale-space maxima – points that represent local maxima in the spatial and scale directions. Implementations have been published describing scale-space maxima detection using the Laplacian [6], Difference of Gaussians [2] and determinant of Hessian [9] functions. These methods commonly use a $3\times3\times3$ non-maximum suppression window to locate scale space maxima. One shortcoming of the method is that it does not account for the fact that 2D maxima generated by the same structure at multiple scales can be separated by several pixels between successive scale levels. The result is that multiple features are detected in some cases where it is not appropriate and hence, scale selection fails in these cases.

Scale-space pyramids are often sub-sampled as scale increases in order to reduce the number of image pixels as the information content is reduced. The purpose is purely to reduce processing cost. The scale-space maxima method places constraints on how a scale-space pyramid can be sampled, since a practical non-maximum suppression algorithm requires that the pyramid levels be sampled with the same resolution. The precision of detection is governed by the sampling density of the pyramid. Three dimensional interpolation can be used to estimate a more precise position for the detected scale-space maxima.

The scale space maxima approach was expanded in [8] by locating points that are local maxima of the Harris function in the spatial dimensions and maxima of the Laplacian in the scale dimension. Utilizing different functions for selecting scale and for locating spatial position makes it possible to combine the functions that are most stable in each domain. The Harris measure produces stable maxima in the spatial domain, but rarely produces stable maxima in scale-space – a local maximum can be displaced a long distance over a change in scale (depending on image structure). The Laplacian contributes the ability to select a stable characteristic scale by functioning like a correlation detector for structure size.

An iterative method was also presented in [8] that adapts each of a set of multi-scale points to a characteristic scale. For every iteration, the algorithm selects scale by searching vertically in the scale space for a maximum in the Laplacian function, followed by relocating the maximum of the Harris function at the chosen scale. This technique is very accurate in selecting scale and is a lot more effective in dealing with unstable spatial localization functions; however it is process intensive due to its iterative nature. The algorithm also results in multiple features converging. Although this is an indication of stability, it also implies that some of the processing effort is made redundant.

III. Scale-Space Feature Locus Analysis by means of Clustering

A method is sought to analyze and describe an image feature over a range of scales, instead of only at characteristic scales. The objective is to properly keep track of a feature as it changes over scale and build a representation that is suited to the dynamic nature of scale-space features with minimum computational cost.

Denote the scale-space representation of an image $I(\mathbf{x})$ as the convolution of the image with the 2D Gaussian smoothing function,

$$L(\mathbf{x},\sigma) = I(\mathbf{x}) * g(\mathbf{x},\sigma).$$

Apply a detection function $D(I)$ to the scale-space image to produce a scale space saliency map,

$$D(\mathbf{x},\sigma) = D(L(\mathbf{x},\sigma)).$$

Select a point in $D(\mathbf{x},\sigma)$ that is a local maximum in the spatial dimensions at a particular scale (it does not need to be a maximum in the scale direction). As the scale dimension is traversed smoothly, the location of this point will vary smoothly. Define the locus of the maximal point in scale-space as $\mathbf{x}_\sigma = f(\sigma)$. The objective of the new method for scale-space feature analysis is to extract the locus $f(\sigma)$ for every image feature using the information available in a set of multi-scale features.

The method developed below for extracting feature loci makes use of the typical behavior of features in scale-space to cluster a set of multi-scale features into graphs that represent discrete feature loci. Before the algorithm itself is described, the typical behavior of features in scale-space is presented.

A. *Feature Behavior in Scale-Space*

The spatial location of image features generally changes as scale changes. The velocity of features with respect to scale depends entirely on the arrangement of physical structures in an image, particularly steep gradients or edges. The behavior of features in scale-space can be described as follows:

- Features move away from edges as scale increases.
- The distance a feature is displaced over a change in scale is related to its proximity to edge structures in the image and the magnitude of scale change.
- In the presence of a single straight edge, the displacement of a maximum of the Laplacian function is in the direction perpendicular to the edge and the velocity is linearly related to the change in standard deviation of the Gaussian operator. The maxima of other functions behave similarly, though the velocity relation may not be exactly linear.
- The velocity of the feature in a corner structure depends on the angle between the two edges forming the corner junction.
- Features can converge as scale increases.
- A feature cannot diverge into multiple features as scale increases, due to the characteristics of the Gaussian function.
- Features related to different structures are by definition separated by edges, and as such will be located a significant distance apart and will propagate in diverging directions, at least until the separating structure is no

longer dominant (affected by larger distant structures at higher scale).
- Converging features accelerate as they approach convergence.

B. The Clustering Algorithm

A pseudo code version of the clustering algorithm is included in Algorithm 1. In this section the algorithm will be described and the relevant design decisions discussed.

Algorithm 1: Clustering Algorithm.

```
Input:
A set of N features with discrete scale levels [0...n].
The search radius multiplier denoted k.
begin
  for i = 0 → N do
    (i) Get the next feature f_i.
    (ii) if feature has already been clustered then
      go to next feature.
    (iii) Create a new cluster C with f_i as first feature.
    (iv) Set σ = scale(f_i) the scale of the feature.
    (v) repeat
      (a) Calculate the search radius r = kσ_i.
      (b) Search for the closest feature at scale
      level σ+1 that is within radius r of the last
      feature. Denote this feature f_new.
      (c) if feature found within search radius then
        if f_new has already been clustered then
          Detected a convergence.
          (1) Duplicate part of the cluster that
          f_new belongs to, starting from f_new
          upwards in scale and append the
          duplicate section to the current cluster
          C.
          (2) Continue at (vi).
        else
          (1) Add f_new to C.
          (2) Set σ = scale(f_new).
        end
      else
        Cluster complete. Continue at (vi).
      end
    until σ = σ_max
    (vi) Add the completed new cluster to the set of
    clusters.
  end
end
Output:
A set of clusters. Each cluster lists a set of features in
order, representing the feature locus.
```

The first step is to detect features using a 2D detector at multiple discrete scales. The type of detector used is not important for the overall structure of the algorithm, as long as a set of features detected at multiple discrete scale levels is produced. Clustering begins by selecting any one of the lowest scale features. Next, the area in the spatial vicinity of the previously selected feature, but one scale level higher, is searched for features. The choice of search radius is discussed later. If a feature is found, it is added to the cluster and the search continues by searching in the vicinity of the new feature, one scale level higher. If no features are found in the designated search area or the highest scale level is reached, then the cluster is considered complete. The next cluster is constructed in the same way by selecting one of the lowest scale features and following the same procedure. This continues until all features have been clustered.

It is possible that multiple features can be found in the search area of one feature near a point in scale-space where multiple features converge. In this case the closest feature is selected to form the next feature in the cluster. There are different possibilities for dealing with converging features. One method is to duplicate the shared part of a cluster where two or more features converge so that there exists a cluster for each of the original features. This approach to dealing with convergence has been chosen and implemented for its simplicity and because it is suitable for the scale selection method described in Section IV-A. Alternatively, it is also possible to use a tree structure to represent clusters that include converging features.

The clusters produced by this algorithm have the following properties:

- The cluster is in the form of a simple graph with each element (feature) relating to no more than two other elements.
- The features in the cluster are ordered according to scale, and are contiguous in discrete scale.
- All the features in the cluster relate to the same image structure.

Figure 1 shows the output of the clustering algorithm. The algorithm was applied to a set of determinant of Hessian features detected in the first image of the “Corridor” image sequence (the full sequence is available online at http://www.robots.ox.ac.uk/~vgg/data/).

C. Choice of Search Radius

The size of the area searched for new features to add to a cluster must be chosen large enough to include all features from the same structure as the current cluster. At the same time features from other structures should be avoided. The choice of search radius depends on the behavior of the particular type of features being clustered; however some basic principles are generally applicable to most detectors. The given clustering algorithm does not attempt to analyze image structures in order to select the feature search area, but depends on knowledge of feature behavior only. The major factors that need to be considered are:

1) The expected feature drift velocity in terms of the number of pixels of displacement vs. the change in scale;
2) How close together features of different structures can be located;
3) How densely the scale-space is sampled.

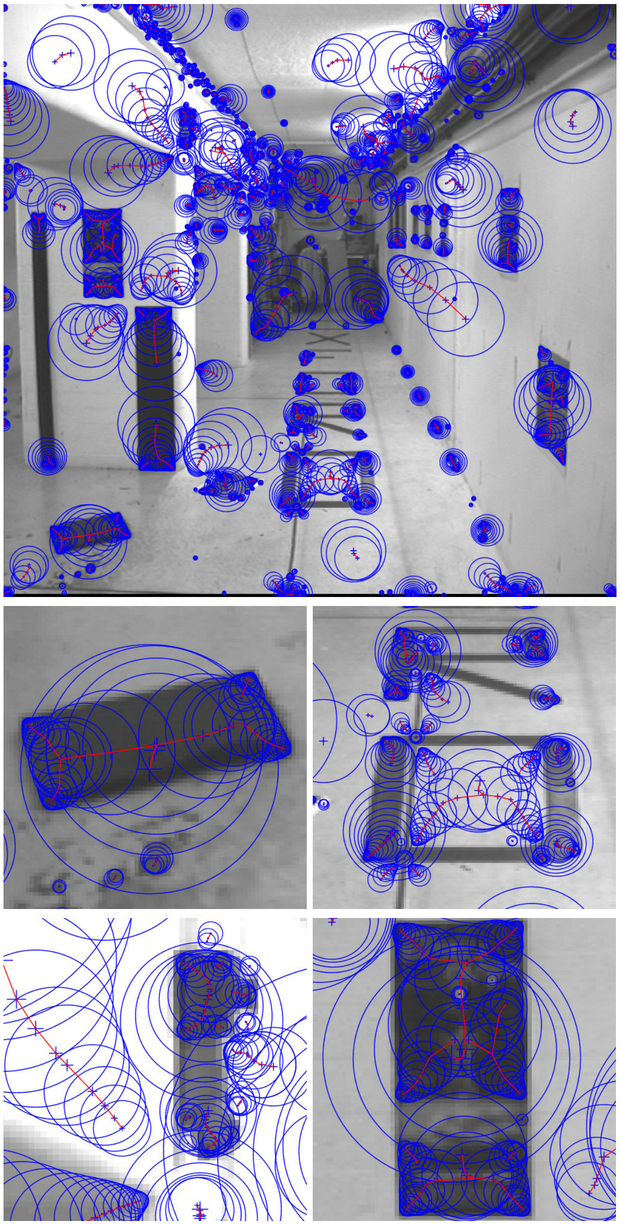

Fig. 1. Sub-regions of the first Image of the "Corridor" sequence, with Clustered Features. Blue circles indicate features and red lines joining circle centers indicate clusters.

Refer to Section III-A for a discussion on general feature behavior. The velocity of features located near corners depends on the angle of the corner, and can be very large where edges are near parallel. Furthermore, converging features accelerate as they approach convergence. Based on these observations, it is desirable to maximize the search radius to improve the likelihood of locating features that belong to the same structure.

The dominant limiting factor in choosing a search radius is the minimum distance that features from different structures can be located from each other. The determinant of Hessian detector is considered as an example; This detector produces maxima at a distance of one standard deviation from an edge. Therefore, features from opposite sides of an edge can be located no less than two standard deviations from each other at any particular scale. The search radius limit for clustering determinant of Hessian features is chosen as $1.1\sigma_i$, where σ_i is the standard deviation of the last feature added to the cluster, to compromise between the search area for a given feature and the search area of a potential neighboring feature at the same scale.

Figure 2 demonstrates the effectiveness of this method for search radius selection. Each image shows a case where a Determinant of Hessian detector is applied to a pyramid with different scale sampling density. The clustering algorithm produces consistent results in all cases.

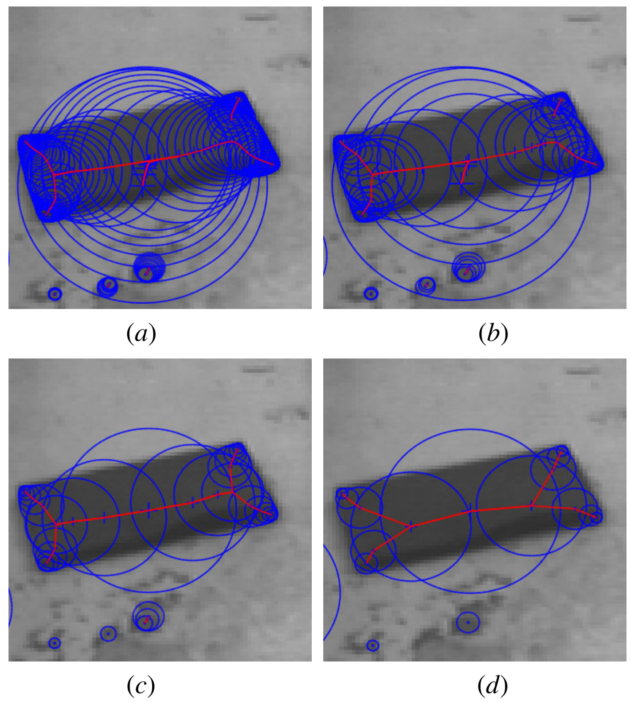

Fig. 2. An Image with Clustered Determinant of Hessian Features produced from pyramids with different scale sampling densities: (*a*) 8, (*b*) 4, (*c*) 2, (*d*) 1 level(s) per octave of σ.

IV. Applications of Feature Clusters

A. Scale Selection using Clustered Features

A characteristic scale selection method using clusters is presented as a demonstration of the utility of the cluster representation. Scale selection is made very simple by clustering features. The cluster represents the feature locus $\mathbf{x} = f(\sigma)$. Denote the scale response function for the image scale-space (e.g. the scale normalized Laplacian) as $S(\mathbf{x},\sigma)$. For each cluster, simply evaluate a scale response function at each node in the cluster. This produces a 1D scale response function $S_f(\sigma) = S(f(\sigma),\sigma)$. Selecting a characteristic scale for the image structure described by the cluster is then simply a matter of analyzing $S_f(\sigma)$. Points chosen on $S_f(\sigma)$ directly translate to selected features in the cluster. Figure 3 shows an example of the scale response function of cluster of Determinant of

Hessian features. Characteristic scale selection simply consists of selecting the features at the local maxima in the function.

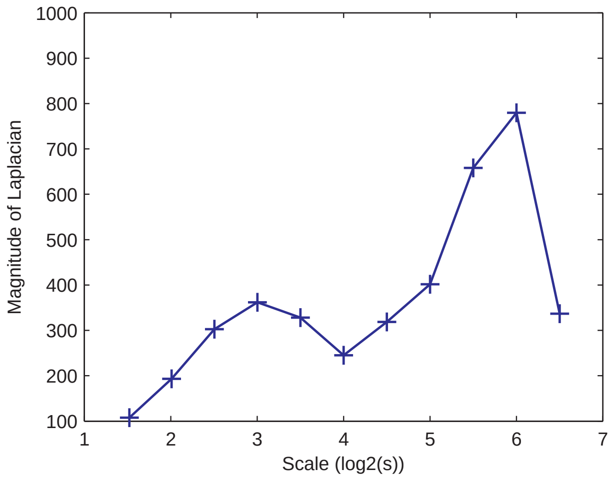

Fig. 3. Laplacian Response of a Feature Locus

Using clusters for scale selection has several advantages over other methods, including:

- Interpolation can be performed by interpolating the 1D feature scale response function (for example, using a parabolic fit) and separately interpolating the x and y positions of the feature as 1D functions of scale. This is somewhat simpler than interpolating a position in 3D scale-space.
- If a very high degree of accuracy is desired, an iterative technique could make use of the cluster information to select a good starting position and to guide sampling of the scale space.
- The clustering algorithm only needs to operate on features and not necessarily on the scale-space pyramid directly. The scale-space pyramid may therefore be sampled in any fashion without causing practical difficulties in the scale selection process. This has very significant implications for accelerating both pyramid construction and multi-scale peak detection.

B. Detector Processing Cost Comparison

Three Determinant of Hessian based detectors were implemented and their processing time compared to test the performance of cluster analysis:

1) A simple scale-space maxima method using 3D non-maximum suppression, similar to .
2) A cluster based method using 2D non-maximum suppression in the spatial dimensions, the described clustering technique and scale selection by selecting maxima in the scale response functions derived from clusters.
3) A cluster based method using a pyramid that utilizes progressive sub-sampling of pyramid levels to increase efficiency.

Note that in this case the Determinant of Hessian function is used for both spatial and scale localization, in order to simplify the scale-space maxima method and to have consistency in the scale selection method used for all three detectors. The cluster based detectors could easily use a different function for scale localization in combination with using the Determinant of Hessian function for spatial localization. All that is required is that the scale function be evaluated at each feature location.

Similar efficient non-maximum suppression techniques were used for 2D and 3D maximum detection. The 3D version of the technique requires only a fraction more computation time to scan a scale-space pyramid than the 2D version. All implementations involved double precision floating point arithmetic. Detectors were applied to the "graffiti" sequence found at http://www.robots.ox.ac.uk/~vgg/data/ and detector times and feature counts were averaged across the sequence.

Detector	**Time (ms)**	**Features**
Scale-space Maxima	1451.33	3006.83
Clustered	1581.75	2781.17
Clustered (sub-sampled)	536.98	2178.83

TABLE I
AVERAGED DETECTION TIMES AND FEATURE COUNTS FOR DETECTORS APPLIED TO THE "GRAFFITI" SEQUENCE.

Average processing times and output feature counts are presented in Table I. It can be seen that the clustered approach requires a relatively small amount of extra processing time and the total time is comparable to scale-space maxima when applied to the same image pyramid. The clustered approach also produces approximately 10% less features than the scale-space maxima method – this is explained in the discussion of Figure 4 in the following paragraph. The detector using clustering as well as progressive sub-sampling of the image pyramid completes in a much shorter time than the other detectors. This shows that large performance gains can be achieved by progressively sub-sampling the levels of the pyramid without any modification to the 2D maxima detector or the clustering algorithm. The lower number of features detected when the pyramid is sub-sampled is a result of the loss of information and aliasing due to sub-sampling.

C. Visual Comparison of Detectors

Figure 4 shows sections of the output of the scale-space maxima detector and the clustering based method from Section IV-B applied to various images. Notice that in some circumstances the scale-space maxima detector produces multiple features for a structure where the feature translates a large distance over a change in scale. The cluster based method does not suffer the same problem, as it is able to track the feature over scale and select only features that produce a maximum scale response. The result is that fewer low quality features are produced. The last row of images in Figure 4 show the output of an image section with a large amount of detail. Notice that in this case, the scale-space maxima detector does not produce as many unwanted features, and the two detectors produce similar output. Also notice that the clustering method recovers more unique features in a few high detail areas, such as the areas marked by red circles in the last line of images.

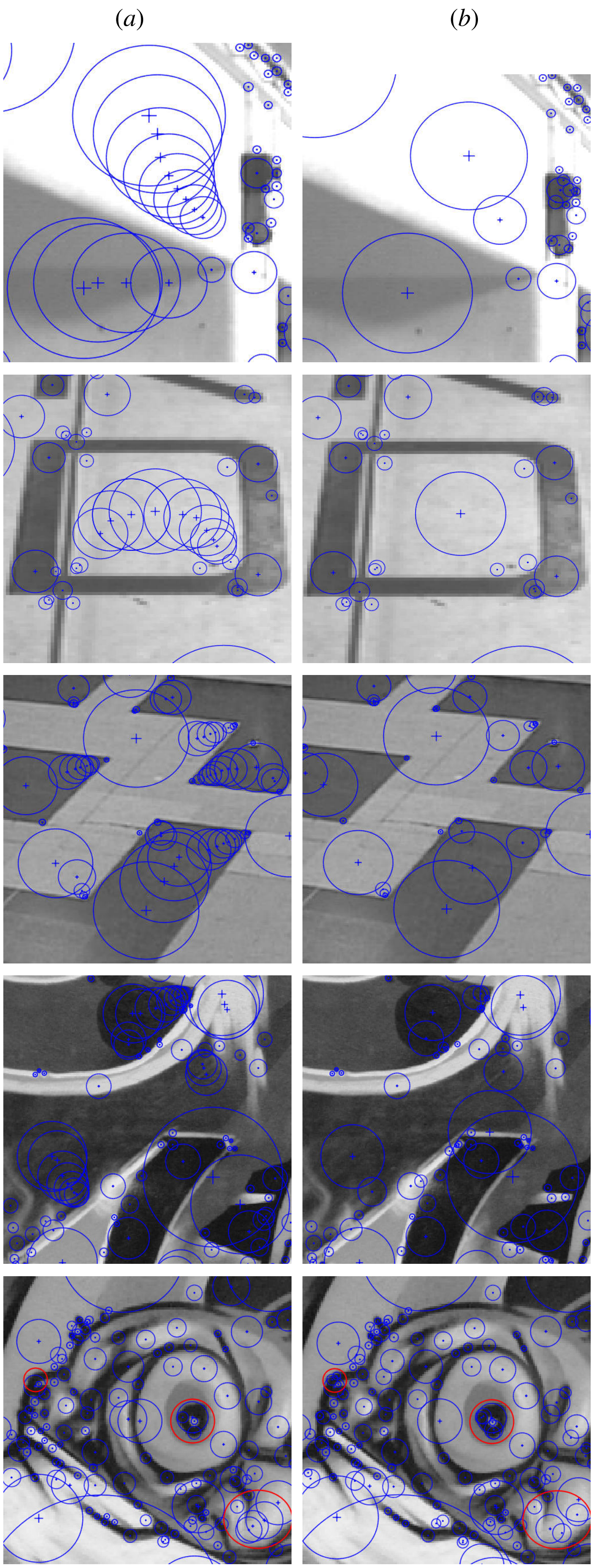

Fig. 4. Column (*a*): Features produced by scale-space maxima detection. Column (*b*): Features produced by clustered scale selection.

D. Repeatability Tests

The repeatability test described in [10] was used to compare the repeatability performance of the cluster based Hessian detector, the scale-space maxima detector and the Hessian Laplace detector [10] (an iterative technique using the Hessian for spatial localization and the Laplacian for scale localization). The repeatability test projects features detected in a test image to a base image using a ground truth homography. The amount of overlap between each base image feature and each projected feature is then measured and correspondences determined according to a minimum overlap threshold. The repeatability rate is calculated as the number of correspondences as a percentage of the minimum of the number of features in each image. The repeatability test software, test data and Hessian Laplace detector software were acquired from http://www.robots.ox.ac.uk/∼vgg. The "bark" and "boat" sequences were used for this test, as they involve scale change.

The results of the repeatability tests are presented in Figures 5 and 6. It can be seen that the clustering based method consistently achieves higher repeatability than the the scale-space maxima method, while producing only a fraction fewer correspondences. The more process intensive Hessian Laplace detector still outperforms the other detectors on average in terms of repeatability and produces more correspondences. It is worth noting that the Hessian Laplace detector makes no effort to eliminate duplicate features arising from the convergence of multiple features, which may be contributing to both repeatability and the number of correspondences. The fact that the clustered approach achieves higher repeatability than Hessian Laplace in certain cases indicates that it may provide a better initial starting point for the Hessian Laplace's iterative method.

E. Other Potential Applications

A major potential use of clustered features representation is in reducing excessively large feature sets to a smaller set of features that are more stable. Image features are often used to form a compact description of the contents of an image. If, however, a multi-scale analysis of the scene results in a very dense set of features, the feature based representation loses its efficiency. Even after scale selection a feature set may still be too dense. Changing detection thresholds gives a coarse method to control feature density; however, selecting a threshold automatically is not always effective. Feature locus clusters afford more powerful and relevant options in selecting features.

The clustered feature representation's utility for feature selection lies in that it establishes a meaningful structure based relation between multi-scale features. It is possible to reduce a cluster of features to a smaller representation, for example by performing characteristic scale selection as in Section IV-A or by treating the entire cluster as a single feature. Other possibilities include selecting only features based on scale criteria, such selecting features that are stable over a minimum number of scale levels, or that exist over a specific range of scales.

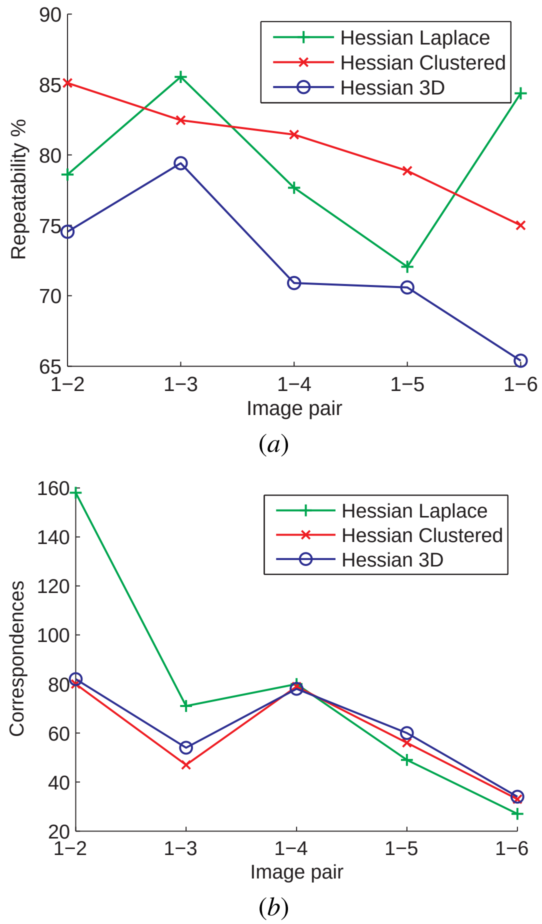

Fig. 5. Repeatability Test using “bark” sequence and 30% overlap error. (*a*) Repeatability Measure. (*b*) Number of Correspondences.

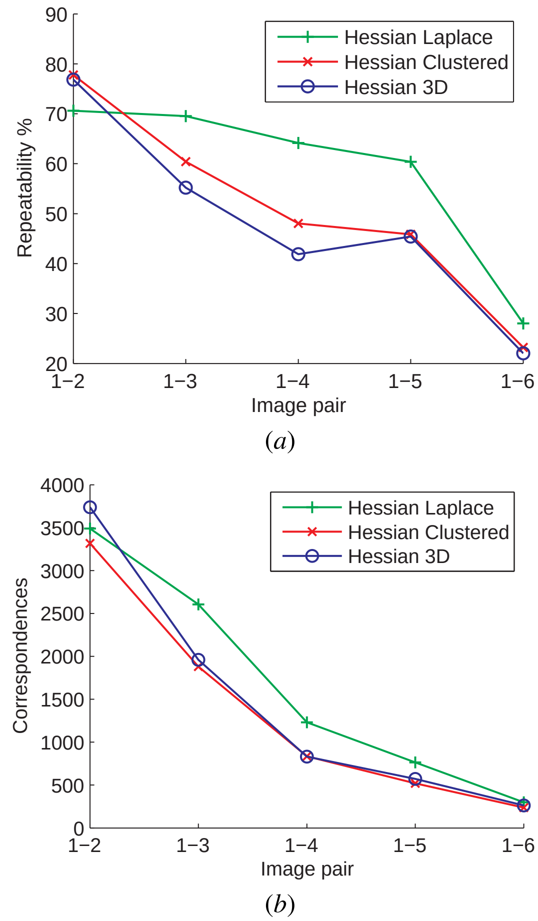

Fig. 6. Repeatability Test using “boat” sequence and 30% overlap error. (*a*) Repeatability Measure. (*b*) Number of Correspondences.

V. Conclusion

A scale-space analysis method has been presented that can describe the loci of image features in scale-space in the form of discrete multi-scale feature graphs or clusters. The method is successful in producing the clusters in an efficient manner from a set of multi-scale features, without the need to interrogate the image data directly. Locus based feature clusters provide a powerful means of feature description, analysis and selection in that they represent meaningful structure based relationships between multi-scale features. Tasks such as characteristic scale selection and filtering features according to specific scale requirements are made simple.

The example application of feature clusters in automatic scale selection demonstrates the use of clusters to improve on simple scale selection methods. The cluster based approach demonstrates a more refined capability to select features from a set of multi-scale features, without the need for an expensive iterative method and without restrictions on the construction of a scale-space representation of the image.

VI. Acknowledgments

This project was supported by the Australian Government Department of the Prime Minister and Cabinet.

References

[1] J. L. Crowley and A. C. Parker, “Representation for shape based on peaks and ridges in the difference of low-pass transform,” Carnegie Mellon University, Tech. Rep. CMU-RI-TR-83-04, May, 1983 1983.

[2] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[3] P. Beaudet, “Rotationally invariant image operators,” in *In International Joint Conference on Pattern Recognition*, Kyoto, Japan, 1978, pp. 579–583.

[4] C. Harris and M. Stephens, “A combined corner and edge detector,” in *Alvey Vision Conference*, 1988, pp. 189–192.

[5] T. Lindeberg, *Scale-Space Theory in Computer Vision*, 1st ed. Stockholm, Sweden: Kluwer Academic Publishers Boston / London / Dordrecht, 1994.

[6] T. Lindeberg, “Feature detection with automatic scale selection,” *International Journal of Computer Vision*, vol. 30, no. 2, pp. 77–116, 1998.

[7] Y. Dufournaud, C. Schmid, and R. Horaud, “Image matching with scale adjustment,” *Computer Vision and Image Understanding*, vol. 93, no. 2, pp. 175–194, 2004.

[8] K. Mikolajczyk and C. Schmid, “Scale and affine invariant interest point detectors,” in *IJCV*, vol. 60, 2004, pp. 63–86.

[9] H. Bay, T. Tuytelaars, and L. Van Gool, “Surf: Speeded up robust features,” in *Computer Vision – ECCV 2006*, Graz, Austria, 2006, pp. 404–417.

[10] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, “A comparison of affine region detectors,” *International Journal of Computer Vision*, vol. 65, no. 1-2, pp. 43–72, 2005.