Finding stable salient contours

Yuxuan Lan, Richard Harvey and Jose Roberto Perez Torres
School of Computing Sciences
University of East Anglia
Norwich, NR4 7TJ, UK
{yl,rwh,jrpt}@cmp.uea.ac.uk

Abstract

Methods for generating maximally stable extremal regions are generalised to make intensity trees. Such trees may be computed quickly, but they are large so there is a need to select useful nodes within the tree. Methods for simplifying the tree are developed and it is shown that standard confidence tests may be applied to regions identified as parent and child nodes in the tree. These tests provide a principled way to edit the tree and hence control its size. One of the algorithms for simplifying trees is able to reduce the tree size by at least 90% while retaining important nodes. Furthermore the tree can be parsed to identify salient contours which are the parameters of stable salient regions.

1 Introduction

Matas *et al* [6] describe an algorithm for extracting *maximally stable extremal regions*. Such regions are claimed to have good performance compared to other feature detectors [3, 7] so form a natural first stage for stereo matching [6], image retrieval [11] and other things. In [6] an image is defined as a mapping $Z^2 \rightarrow S$ and regions are defined as two-dimensional connected sets with four connectivity. A more general notation comes from graph morphology [12], in which arbitrary arrays of pixels may be described via a graph $G = (V,E)$ where V is the set of vertices that labels the pixels, and E is the set of edge sets that define the pixel neighbours. Thus an image may be defined in any number of dimensions and with arbitrary connectivity. The image intensity is a function over V, so that, for example $f(x), x \in V$, is the intensity of pixel x. The set of connected subsets of scale s, a positive integer, containing pixel x is defined as $\mathscr{C}_s(G,x) = \{\xi \in \mathscr{C}_s(G) | x \in \xi\}$, where $\mathscr{C}_s(G)$ is the set of all connected sets in G with s pixels. Graph morphology operators may be applied over such sets. Thus $\psi_s, \gamma_s, \mathscr{M}_s, \mathscr{N}_s : Z^V \rightarrow Z^V$ may be defined, for each $s \geq 1$ as

$$\begin{aligned} \psi_s f(x) &= \max_{\xi \in \mathscr{C}_s(G,x)} \min_{u \in \xi} f(u), \\ \gamma_s f(x) &= \min_{\xi \in \mathscr{C}_s(G,x)} \max_{u \in \xi} f(u), \end{aligned}$$

and

$$\mathscr{M}_s = \gamma_s \psi_s, \qquad \mathscr{N}_s = \psi_s \gamma_s.$$

Thus $\mathscr{M}_s$, in grey-scale morphology, is an opening followed by a closing, both of size s for images defined in any finite dimensional space. The M- and N-sieves [1] of a function,

BMVC 2005 doi:10.5244/C.19.2

$f \in Z^V$ are defined as sequences $(f_s)_{s=1}^{\infty}$ with: $f_1 = \mathscr{M}_1 f = f$, and $f_{s+1} = \mathscr{M}_{s+1} f_s$; $f_1 = \mathscr{N}_1 f = f$, and $f_{s+1} = \mathscr{N}_{s+1} f_s$; both for integers, $s \geq 1$ which are cascades of types of alternating sequential filter. The opening and closing sieves may be defined similarly as cascades of ψ_s and γ_s. Non-zero regions in the differences between successive stages are described in [1] as *granules* as an analogy with granulometries. Theorem 6.36 of [1] proves that if pixels that are neighbours, $\{x, y\} \in E$, with intensities $f_s(x), f_s(y)$, $\delta_s = f_s(y) - f_s(x)$, then $\delta_1 \geq \delta_2 \geq \cdots \geq 0$ or $\delta_1 \leq \delta_2 \leq \cdots \leq 0$, from which it is concluded that sieves preserve scale-space causality. A further consequence is that the *extremal regions* in [6], are granules. An extremal region is defined as a connected set such that all pixels not on the boundary are either all brighter than or all darker than those on the boundary. A set of extremal regions can therefore be formed as the union of the set of extremal regions from an opening sieve and the set of extremal regions from a closing sieve. However it is known that $\mathscr{M}$ and $\mathscr{N}$ trees are more robust to noise and occlusion, probably because they remove maxima and minima in one pass and thus approximate a median filter [1] so there is possible enhancement to [6] by switching to the $\mathscr{M}$ or $\mathscr{N}$ operators, and redefining an extremal region to be a connected set such that all intensities in the set are more extreme than those of the neighbours.

Efficient algorithms exist for the computation of sieves that are similar to [6] or [13] which provide a dense set of nested regions. The relationship between the regions may be described via a tree. The tree has a root node that represents the whole image and children that represent the extremal regions, ordered by scale. Each node in the tree represents a connected grey-level set in the image. A tree edge indicates containment, so that child regions are subsets of their parents. Figure 1 (left) shows a tree computed for a stylised 100×100 pixel image. Here the root node is node 5 and represents the whole image. It has one child, node 4, representing the face, with three children representing the mouth and two eyes. Each tree node has been plotted above the centroid of its associated region and at a height proportional to its depth in the tree. Even though such trees can be built efficiently [13], there are potentially very many nodes in the tree and many of them are associated with noise or fine details. This is illustrated in Figure 1 (right) which is a tree of the same image after the addition of Gaussian random noise ($\mu = 0$, $\sigma = 0.05$)[1]. The

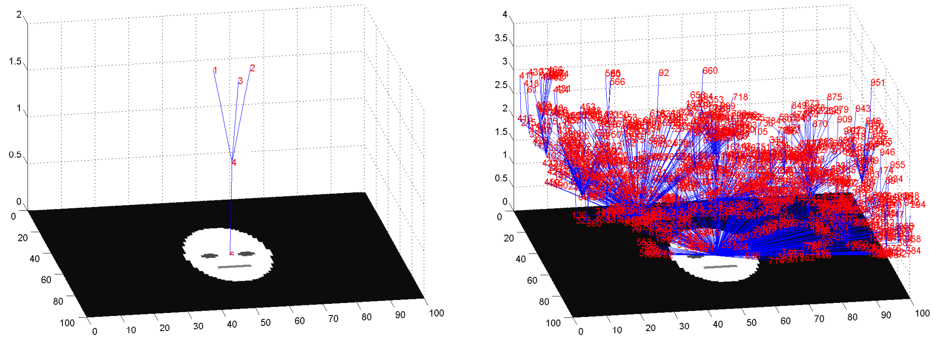

Figure 1: Trees built from original image (left); noisy version (right)

[1] subsequently the image has been quantised to integers in [0,255].

original five nodes are no longer visible within the 958 nodes and the tree has become unpleasantly complicated.

This problem was addressed in [2] in which nodes were removed through two heuristics called *collapsing* or *pruning*. In collapsing, long chains of nodes that have only one child, are replaced by a single node and, in pruning, children are removed by thresholding their area or intensity relative to their parents. These methods are rather unprincipled but sometimes work well. In [6] a variant is introduced in which, for the ith node,

$$q(i) = \frac{|Q_{i+\delta} \setminus Q_{i-\delta}|}{|Q_i|}, \tag{1}$$

where Q_i is the ith connected set in a sequence of nested extremal regions: $Q_1 \subset Q_2 \subset Q_3 \cdots Q_i \subset \cdots$, $|\cdot|$ denotes the number of pixels in the set and δ is a parameter of the method. The idea is to examine the sequence of $q(i)$ to retain nodes that have a local minimum $q(i)$. Such regions are called *maximally stable extremal regions* or MSERs. Because the nodes form a tree, there is a potential uncertainty in [6] because each node might form a part of several sequences of nested extremal regions – nodes may have several children. In Figure 1 for example, $\{1,4,5\}$, $\{2,4,5\}$ and $\{3,4,5\}$ are all nested sequences.

In [8], which also examines stereo matching, it is argued that collapsing and pruning are *ad hoc* methods and are better replaced by a system in which nodes are labelled with statistical significance scores derived under an assumption that the distribution of intensities within a region may be modelled by unimodal Gaussian distributions (denoted here as the difference of Gaussians, or DG, test). The child region consists of the support region for the child and all its children; the parent region is the support region for the parent and all its children excluding the child currently under consideration. Using a standard hypothesis test for the difference of means of two Gaussian distributions of unequal variance ([9], for example) gives a χ^2-score which is then thresholded to allow the progressive removal of nodes. However the unimodal Gaussian assumption is erroneous so, to avoid unrealistically low variance estimates, a Shepperd's correction is used, quantising χ^2, which means that is difficult to remove nodes progressively.

We note some minor differences between the methods so far described. In [6], $q(i)$ is derived from regions above and below the ith node in the tree, whereas in [2] and [8], the authors appear to consider only parent-child relationships. In [6] there is a parameter δ which is difficult to choose correctly for all regions (large δ is appropriate for high contrast blurred objects whereas small δ is appropriate for low contrast or sharper objects). [2] also has some user-defined parameters to control the degree of pruning whereas in [8] the parameter is related to a confidence on statistical model but the model doesn't fit reality very well.

2 Simplification methods

A natural extension is to compute histograms of the intensities in the parent and child regions. To avoid under- or over-smoothed histograms we choose the bin width using the method advocated in [10][2] in which an approximate expression for the bin width is $h = 3.49\sigma n^{-1/3}$. where n is the sample size and σ is the sample standard deviation which, here, is the standard deviation of all the intensities in the image.

[2] The bin width is chosen to asymptotically minimise the integrated mean squared error between the histogram and some true density function.

Two standard hypothesis tests are considered: the χ^2-test for equal distributions and the Kolmogorov-Smirnov test. If the two sets of data are distributed into k bins with p_i, c_i as the counts of the parent and child region pixels in ith bin then, χ^2, is computed as

$$\chi^2 = \sum_{i=1}^{k} \frac{(\sqrt{\frac{\sum_i c_i}{\sum_i p_i}} p_i - \sqrt{\frac{\sum_i p_i}{\sum_i c_i}} c_i)^2}{p_i + c_i} \tag{2}$$

from which the p-value can be calculated. But, because p is monotonically related to χ^2, we need compute only χ^2 which saves time. A large χ^2-value tends to indicate the difference of two distributions.

The Kolmogorov-Smirnov (or K-S) test is designed for continuous data but is easily applied to quantised histograms. If the cumulative histograms corresponding to p_i and c_i are P_i and C_i then the K-S D value is the maximum distance of two cumulative distributions: $D = \max_{-\infty<i<\infty} |P_i - C_i|$. Given the null hypothesis that data sets are drawn from the same distribution, the significance level is approximated by

$$\Pr\{D > \text{observed}\} = Q_{ks}\left([\sqrt{N_e} + 0.12 + 0.11/\sqrt{N_e}]D\right) = Q_{ks}(\lambda) \tag{3}$$

where $Q_{ks}(\lambda) = 2\sum_{j=1}^{\infty}(-1)^{j-1}e^{-2j^2\lambda^2}$, which is monotonic with $Q_{ks}(0) = 1$, $Q_{ks}(\infty) = 0$ where $N_e = n_p n_c/(n_p + n_c)$, and n_p and n_c are the number of pixels in the parent and child regions. As with the χ^2-test, we note that we need compute only λ rather than $Q_{ks}(\lambda)$. More details about χ^2-test and K-S test can be found in, for example, [9].

2.1 Node merging

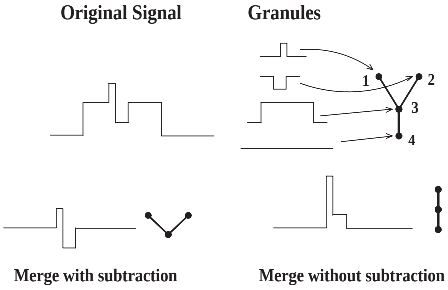

Figure 2: Two ways of node merging.

Removing, or merging, nodes contains a subtlety. As far as the tree goes it is safe to simply remove nodes. However if we wish to reconstruct the image associated with the edited tree then care is needed when editing the associated regions in image. Figure 2 shows, top left, a one-dimensional signal and, top right, its associated components, or *granules*, and tree. Consider removing node 3. On the bottom right, pixels uniquely associated with node 3 have been assigned the intensity of its parent (node 4) and the children (nodes 1 and 2) have been left untouched. Such an editing procedure is unsafe because the tree of the edited image is not the same as the edited tree. The correct alternative, which we use here, is to subtract node 3's intensity from all pixels supported by node 3 (pixels covered by nodes 1, 2 and 3). This procedure gives the correct tree but there is a cost: the resulting image may now contain negative intensities. We overcome this problem by either clipping negative intensities to zero or re-mapping the whole image into a new intensity range while preserving the ordering of the intensities. Of course, if we are interested only in finding regions, such as MSERs or other salient nodes, then these reconstruction details may be ignored.

3 Results

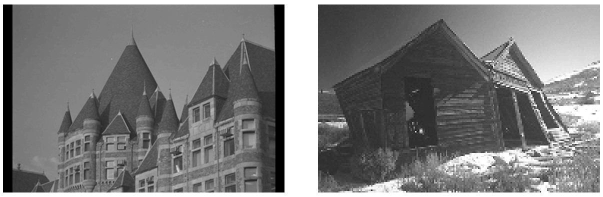

Figure 3: Example images taken from MPEG-7 common colour dataset.

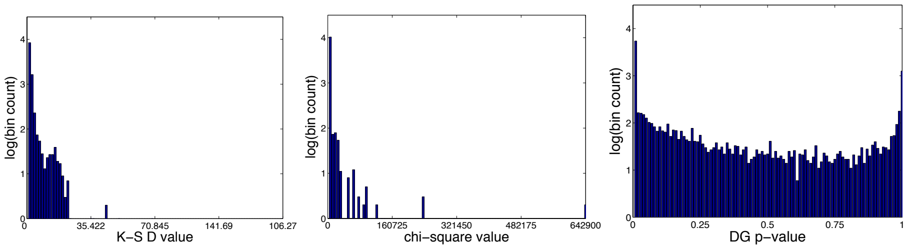

Figure 4: Distribution of K-S D value (left), χ^2 value (middle) and DG *p*-value (right), from the tree built from Figure 3.

Figure 3 shows two images (120×160 pixels[3]) from the MPEG-7 common colour dataset which have trees with 10644 and 12147 nodes. Figure 4 shows the distributions of the scores within a tree. The DG method provides a smoothly varying score whereas the other two methods do not. The top half of Figure 5 shows results from three simplification algorithms with thresholds chosen so that the resulting trees have 10 % of the nodes (hence trees with 1065 and 1215 nodes). Much of the fine detail is removed but the semantic information seems to be intact – subjects are still recognisable. The best performing method is the K-S test (left column), followed by the χ^2 test (centre column) and the DG test (right column) which was the original method. Simplifying still further, gives trees with 533 and 607 nodes, (lower-half of Figure 5), in which the ordering of the methods is unchanged, but now the images are becoming corrupted. In practice we find that it is possible to reduce the tree size to around 5% of its original size without substantially affecting the image. This is illustrated in Figure 6 which shows (left) the original tree and variants with 10% and 5% of the the original tree size.

[3] This paper uses low resolution images for two reasons: firstly the trees are difficult to visualise for large images and, secondly, the $E_{\oplus}$ error used later in this paper is quite slow to compute. In practice computing MSERs using the method described in [6] takes under a second for video frames.

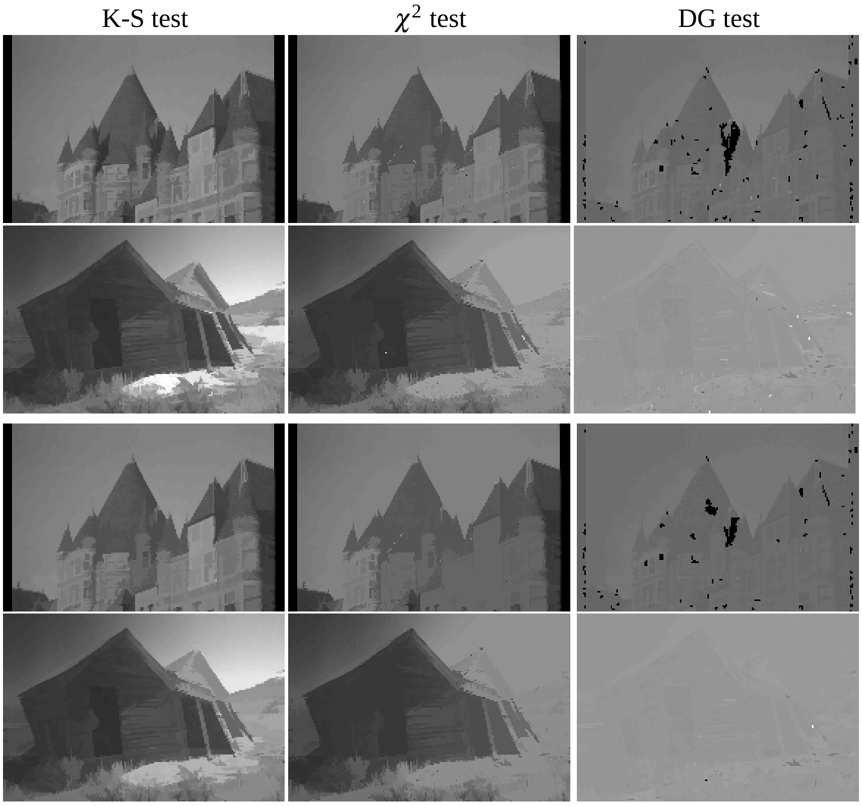

Figure 5: Example images after simplification with various tests. On the top are images reconstructed from trees with 10% of the original number of nodes. On the bottom 5%.

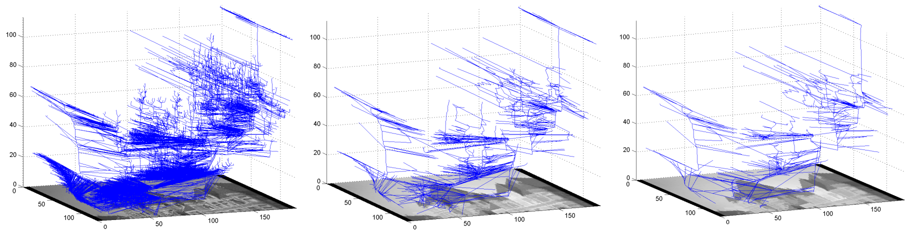

Figure 6: Trees of original image (left of Figure 3) and versions simplified using the K-S test to give 10% and 5% of the original number of nodes.

We can improve on just looking at the images though. Elsewhere [4] we have constructed human segmentations from images in a well-known image database and show that we can measure the effectiveness of a segmenter by comparing automatic segments to the human segments using a normalised error measure based on the binary exclusive-OR between segments. Although the tree is not a segmentation, we can use the same method here. If a tree contains N_T regions, $T_1 \ldots T_{N_T}$ and the ground-truth image contains

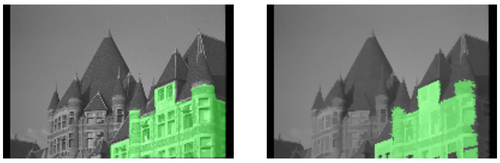

Figure 7: Left: original image with green patch as human segment. Right: a 10% simplified version of by K-S test, with green patch as the node optimally matching the human segment.

N_G regions $G_1, \ldots, G_{N_G}$ then the normalised XOR error is

$$E_{\oplus}(T, G) = \frac{1}{N_G} \sum_{m=1}^{N_G} \min_{n} \frac{|T_n \oplus G_m|}{|T_n| + |G_m|} \tag{4}$$

where $\oplus$ represents a binary exclusive-or. Note that the effect of the min-operator in (4) is to select the best matching tree-node. Figure 7 shows, on the left-hand side, a region identified as important by one of the human labellers and, on the right-hand side, the best matching tree-node identified by the XOR error. The distance between the regions represented by the unedited tree to the ground-truth provides the upper bound that a best simplifying method could possibly reach. Given this distance, a simplified tree can be rejected if the distribution of its distances is too variant from the distribution of distances we get from a full tree. Since these distance distributions are histograms we can, again, use a standard confidence score for comparing histograms. Again then, we optimise the bin size using [10] and apply a χ^2 test and a K-S test, this time on the distance histogram, to obtain Table 1 which shows p-values for the null hypothesis that the two distributions are identical. Thus low p-values tend to support the alternative hypothesis that the two distance distributions are different.

Ratio	K-S		χ^2		DG	
0.05	0.0720	0.0294	0	9.49×10^{-30}	6.65×10^{-7}	1.94×10^{-8}
0.1	0.971	0.872	0.0214	0.0027	0.0045	1.736×10^{-4}

Table 1: p-values for significance tests on the difference of distance histograms. For each tree simplification method we compare the distance histograms using either a χ^2 (left-hand p-value) or a K-S (right-hand p-value)

For a confidence level of 0.01, we can reject H_0 (that the two datasets are not significantly different) for the DG-test on all ratios by both tests. The χ^2 tree simplification method causes us to accept H_0 for the large tree (ratio of nodes 0.1) with the distances compared using the χ^2 scheme. H_0 cannot be rejected for any of the the K-S simplifications which supports the hypothesis that the tree simplified by the K-S test has retained the important nodes in the original tree and is the best method.

So far we have shown that the trees can generate extremal regions of the type defined in [6] and that these trees can be simplified, using a new method, to produce manageable data structures that preserve many important features in the image measured both qualitatively and quantitatively. We now want to parse the trees to produce the equivalent to MSERs. The proposed algorithm draws upon ideas in [6] and [2] in which stable nodes are likely to be drawn from tree branches that are composed of long chains. The algorithm proceeds as follows:

1. All child-to-parent connections are assigned a λ score via (3).
2. The tree is parsed to find monotonic paths (a path in which the depth of nodes is either always increasing or decreasing). Any monotonic path that is longer than a fixed value, l, and has all nodes with λ below a threshold is designated as stable.
3. A stable node is extracted as the middle node of this path.

On the face of it, this algorithm appears rather unsatisfactory because it contains two parameters. However note that $Q_{ks}(\lambda)$ is a properly normalised confidence so thresholding λ amounts to selecting an operating point for a hypothesis test. The length threshold, l, like δ in (1), can be set to give fewer or more stable-contours.

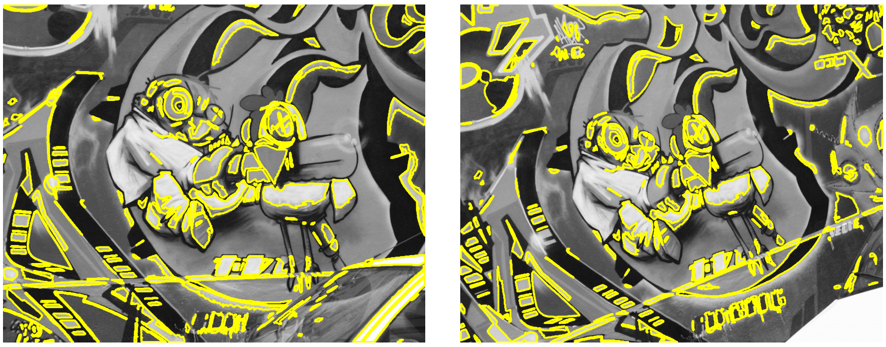

Figure 8: Regions detected in the *Graffiti* image from [7]. Left 0° and right 20° image.

Figure 8 shows example regions detected by the system. Note that, unlike salient region methods based on corners or fixed windows, the regions are the shape of the intensity contours in the images and hence are more likely to represent meaningful regions.

The evaluation of region detectors has recently received some attention [7] so there is now a standard test with Matlab source code that we can use. In [7] the authors present multiple images and ground-truth homographies that would map one into another. A peculiarity of the evaluation is that region is represented by ellipse with identical area moments up to second order. This ellipse is then warped to another image and the test looks for its corresponding ellipse. If the overlap error between these two ellipses is small (below 40% in [7]), a correspondence is achieved. Therefore, the repeatability of a region detector for these two images can be computed as the number of ellipse correspondences divided by the smaller of the numbers of regions generated from each images.

Figure 9 shows the results from two images from the set in [7]. Note that, for these images, the MSER technique is already the best performing so it is gratifying to be able to improve on the best known method. However, as with the δ parameter in the MSER method, a poor choice of k can degrade performance. For interest, we also show an alternative node selection method based on entropy. In [5] it is argued that entropy forms a suitable selection scheme for selecting the scale of window. The idea is to vary a window by scale and select a node which is a local maximum in the scaled entropy. Here we

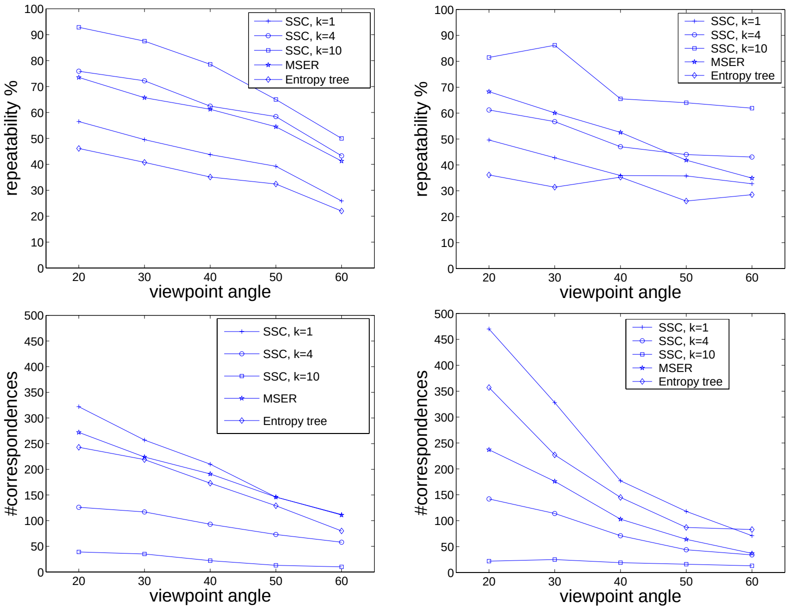

Figure 9: Repeatability (top) and correspondence count (bottom): graffiti images(left) and bikes images (right) from [7] for stable salient contours (SSCs), MSERs and an alternative entropy-based method. The branch length threshold $l = 0.02 \times k \times$ treedepth.

compute, for the ith node $H_i = -\sum_{n=1}^{255} p_{ni} \log_2 p_{ni}$ where p_{ni} is the probability of the nth grey-level in the ith node. Leaf nodes are assigned zero entropy. We identify salient nodes as those that have locally maximal entropy on monotonic paths. As in [7] we find that entropy is not a very effective measure. Possibly we could, as in [5], scale the entropy by $s^2(\sum_{n=1}^{255} |p_{ni} - p_{nj}|)/(2s-1)$ where j is a child of node i, but for large trees this is an expensive operation.

4 Conclusions

Sieve trees are well known decompositions of images [1]. They generate a hierarchy of connected sets that is potentially very large (the trees have many nodes). This paper has introduced two new methods for simplifying the trees. The method based on the Kolmogorov-Smirnov test is easy to use and is the best performing. It is possible to reduce the tree to 10% of its original size without affecting the image too much. By comparing the nodes in a full tree and a simplified tree with ground truth from human segmenters we cannot reject the hypothesis that difference between a simplified tree and ground truth is identical to the difference between full tree and ground truth – implying that the simplification method retains the important regions.

This paper has further demonstrated that stable salient contours may be generated from sieve trees. These salient contours are related to, but are not the same as, Maximally Stable Extremal Regions (MSERs) from [6]. We evaluate these stable salient contours using the standard methodology in [7]. We find that the new methods can perform as well as MSERs which themselves are known to be among the best performing methods. However, as with MSERs, there is a dependence on a parameter, k, which is related to the length of stable branches. For future work we hope to eliminate this parameter. For the left image in Figure 3, the efficient sieve tree algorithm can build the tree within 0.5 second on a computer with Pentium III processor and 522MB RAM. However, the current implementation of the full algorithm is rather slow and inefficient, mainly due to the exhaustive search of all possible paths in the tree.

References

[1] J. A. Bangham, R. W. Harvey, and P. D. Ling. Morphological scale-space preserving transforms in many dimensions. *Journal of Electronic Imaging*, 5(3):283–299, July 1996.

[2] J. A. Bangham, J. R. Hidalgo, R. C. Harvey, and G. C. Cawley. The segmentation of images via scale-space trees. *Proc. British Machine Vision Conference.*, pages 33–43, 1998.

[3] F. Fraundorfer and H. Bischof. Evaluation of local detectors on non-planar scenes. In *Proc. of the 28th Workshop of the Austrain Association for Pattern Recognition*, pages 125–132, Österreichische Computer Gesellschaft 3-85403-179-3, Hagenberg, 2004.

[4] S. Gibson, J. A. Bangham, and R. Harvey. Evaluating a colour morphological scale-space. In *Proc. British Machine Vision Conference*, pages 799–808, 2003.

[5] T. Kadir, A. Zisserman, and M. Brady. An affine invariant salient region detector. In *Proc. European Conference on Computer Vision*, pages 257–269. Springer, 2004.

[6] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In P. Rosin and D. Marshall, editors, *Proc. British Machine Vision Conference*, volume 1, pages 384–393, 2002.

[7] 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*, 2005. to appear.

[8] K. Moravec, R. W. Harvey, and J. A. Bangham. Scale trees for stereo vision. *IEE Proc. Vision Image and Signal Processing*, 147(4):363–370, 2000.

[9] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C: the art of scientific computing.* Cambridge University Press, 1992.

[10] D. W. Scott. On optimal and data-based histograms. *Biometrika*, 66(3):605–610, 1979.

[11] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proc. International Conference on Computer Vision*, pages 1470–1477, Oct 2003.

[12] L. Vincent. Graphs and mathematical morphology. *Signal Processing*, 16:365–388, 1989.

[13] L. Vincent. Fast opening functions and morphological granulometries. In *SPIE*, volume 2300, page 253:267, 1994.