Edge Foci Interest Points

C. Lawrence Zitnick
Microsoft Research
Redmond, WA
larryz@microsoft.com

Krishnan Ramnath
Microsoft Research
Redmond, WA
kramnath@microsoft.com

Abstract

In this paper, we describe an interest point detector using edge foci. Unlike traditional detectors that compute interest points directly from image intensities, we use normalized intensity edges and their orientations. We hypothesize that detectors based on the presence of oriented edges are more robust to non-linear lighting variations and background clutter than intensity based techniques. Specifically, we detect edge foci, which are points in the image that are roughly equidistant from edges with orientations perpendicular to the point. The scale of the interest point is defined by the distance between the edge foci and the edges. We quantify the performance of our detector using the interest point's repeatability, uniformity of spatial distribution, and the uniqueness of the resulting descriptors. Results are found using traditional datasets and new datasets with challenging non-linear lighting variations and occlusions.

1. Introduction

Identifying local features is a critical component to many approaches in object recognition, object detection, image matching and 3D reconstruction. In each of these scenarios, a common approach is to use interest point detectors to estimate a reduced set of local image regions that are invariant to occlusion, orientation, illumination and viewpoint changes. The interest point operator defines these regions by their spatial locations, orientations, scales and possibly affine transformations. Descriptors are then computed from these image regions to find reliable image-to-image [19, 23] or image-to-model matches [5, 22]. It is desirable that a good interest point detector has the following three properties: (1) The interest points are repeatable, (2) the descriptors produced from them are unique and (3) they are well-distributed spatially and across scales.

An interest point is defined based on some function of the image, typically a series of filtering operations followed by extrema detection. Some of the techniques that work based on this principle are the Harris corner detector [7], the Difference of Gaussian DoG detector [11], the Laplacian detector [10], and their variants including Harris-Laplace [14, 16] and Hessian-Laplace detectors [17]. Detectors that find affine co-variant features [17] have also been proposed such as Harris-affine [1, 15], Hessian-affine [17], Maximally Stable Extremal Regions MSER [13] and salient regions [8]. Most of these approaches perform a series of linear filtering operations on the image's intensities to detect interest point positions. However, filtering intensities directly can result in reduced repeatability under non-linear lighting variations that commonly occur in real world scenarios. Furthermore, when detecting objects in a scene, changes in the background will also result in non-linear intensity variations along object boundaries, resulting in a similar reduction in repeatability.

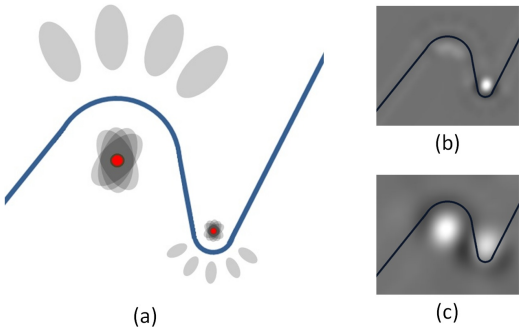

Figure 1. Illustration (a) of the position of edge foci (red dots) relative to edges (blue line). Grey ellipses show the area of positive response for the orientation dependent filters. Peak aggregated filter responses for a small scale (b) and large scale (c).

In this paper, we propose detecting interest points using edge foci. We define the set of edge focus points or edge foci, as the set of points that lie roughly equidistant to a set of edges with orientations perpendicular to the point, shown as red dots in Figure 1(a). The detection of edge foci is computed from normalized edge magnitudes, and is not directly dependent on the image's intensities or absolute gradient magnitudes. Compared to image intensities, we hypothesize that the presence of edges and their orientations is more robust to non-linear lighting variations and background clutter [18, 4, 25]. Edge foci are detected by applying different filters perpendicular and parallel to an edge.

The filter parallel to the edge determines the edge's scale using a Laplacian of a Gaussian. The second filter blurs the response perpendicular to the edge centered on the predicted positions of the foci, shown as grey ellipses in Figure 1(a). Aggregating the responses from multiple edges results in peaks at edge foci. Figures 1(b,c) show examples of two detected foci at different scales.

As described in Section 3, our detector has three stages. First, we compute locally normalized edge magnitudes and orientations. Second, we perform orientation dependent filtering on the resulting edges, and aggregate their results. Finally, we find local maxima in the resulting aggregated filter responses in both spatial position and scale. In Section 4, experimental results show the increased repeatability of our detector and its competitive performance on real world tasks. Given the non-linear operations performed by the detector, our approach does require increased computational resources over more traditional approaches [7, 11]

2. Previous work

Interest point detectors can be categorized based on the transformations for which they are co-variant. The earliest interest points were corner detectors [7] that detected the positions of corner-like features in the image. Scale covariant features were later introduced by Lindeberg [10] and popularized by Lowe [11] using Laplacian or Difference of Gaussian filters. Recently, several interest points detectors have been proposed that are co-variant with affine transformations. Matas et al. [13] detects stable regions of intensity, while Kadir et al. [8] detects salient regions. Combinations of either Harris or Hessian corner detectors, followed by Laplacian scale selection and affine fitting were proposed by Mikolajczyk et al.[15] and Baumberg et al. [1]. A comparison of the affine co-variant interest point detectors can be found in [17]. Computationally efficient detectors have also been proposed, including SURF [2], FAST [21] and CenSurE [12].

The use of edges for interest point detection has received less attention. Mikolajczyk et al. [18] proposed an interest point detector that finds points equidistant from edges. Unlike our approach, they do not consider edge orientation in their filter response, resulting in numerous peaks that are not well localized. Mikolajczyk et al. [17] also proposed an affine co-variant edge-based region detector using Harris corners [7] to locate positions. Canny edges [4] determined the remaining affine parameters. However, it underperformed the authors' other detectors.

3. Approach

In this section, we describe our approach to interest point detection. We begin by describing the computation of normalized edges and orientations (Figure 2(b)), followed by orientation dependent filtering (Figure 2(c,d)). Finally, we discuss approaches to computing the filter responses across scales, and the detection of maxima in scale space.

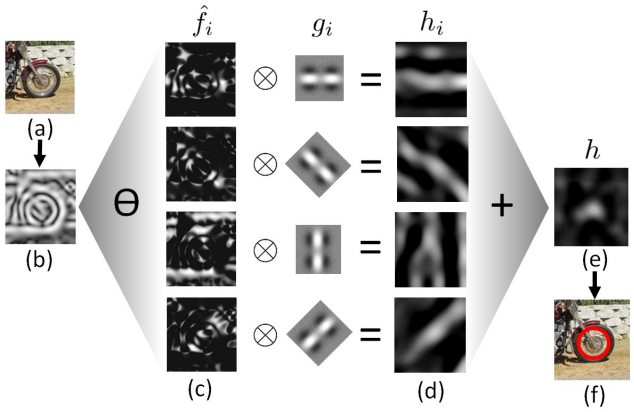

Figure 2. Flow diagram of the detector: (a) input image, (b) normalized gradient $\hat{f}$, (c) normalized gradients separated into orientations $\hat{f}_i$, (d) responses after applying oriented filter $h_i = \hat{f}_i \otimes g_i$, (e) the aggregated results h, and (f) detected interest point.

Initially, we restrict our discussion to finding interest points at a particular scale σ in image I. The scale σ defines the size of the local structures to be detected, i.e. the distance between the edges and their foci, Figure 1.

3.1. Computing normalizing edges

The first stage of our approach computes normalized edge magnitudes and their corresponding orientations for an image I. Similar to other Gaussian pyramid based techniques [11, 9], we first blur the image using a Gaussian kernel. Assuming we are finding interest points at a particular scale σ, we blur using a standard deviation of $\alpha\sigma$ to ensure the detector is scale covariant, where $\alpha = 0.25$ for all experiments. It is important that α is large enough to remove quantization noise when computing orientations at smaller scales, while not being too large that it blurs image structures. The intensity of a pixel p in the blurred image I^σ at location (x_p, y_p) is denoted $I^\sigma(p)$ or $I^\sigma(x_p, y_p)$. The horizontal gradient $I^\sigma_x(p)$ of the pixel is equal to $I^\sigma(x_p + 1, y_p) - I^\sigma(x_p - 1, y_p)$ and similarly for the vertical gradient $I^\sigma_y(p)$. The magnitude of the gradient for pixel p is the Euclidean norm of its gradients, $f(p) = \|[I^\sigma_x(p)\ I^\sigma_y(p)]^{\mathrm{T}}\|_2$. The orientation is defined as $\theta(p) = \arctan(I^\sigma_y(p)/I^\sigma_x(p))$. We assume the orientations are not polarized, i.e. $\theta(p) \in [0, \pi]$.

If the original image already exhibits some spatial blurring with a standard deviation of σ_0, the amount of blur should be reduced to $\sqrt{(\alpha\sigma)^2 - \sigma_0^2}$ to ensure a resulting blur with a standard deviation of $\alpha\sigma$. In our experiments we assume an initial blur of $\sigma_0 = 0.5$, since most real images exhibit some fractional pixel blur.

To normalize the edge magnitudes, we use the same approach as proposed in [25]. First, we compute the average gradient in a local spatial neighborhood of each pixel. The average Gaussian weighted gradient $\bar{f}(p)$ in a neighborhood

N of p is:

$$\bar{f}(p) = \sum_{q \in N} f(q)\mathcal{G}\left(q - p; \alpha\sigma\sqrt{(\lambda^2 - 1)}\right), \quad (1)$$

where $\mathcal{G}(x; s)$ is a normalized Gaussian evaluated at x with zero mean and a standard deviation of s. We set $\lambda = 1.5$ for all our experiments. Next, we divide f by the mean gradient $\bar{f}$ to compute our normalized gradient $\hat{f}$:

$$\hat{f}(p) = \frac{f(p)}{\max(\bar{f}(p), \epsilon/\sigma)}, \quad (2)$$

where $\epsilon = 10$ is used to ensure the magnitude of $\bar{f}(p)$ is above the level of noise. An example of the normalized gradients is shown in Figure 2(b).

3.2. Orientation dependent filtering

Our next stage computes a series of linear filters on the normalized gradients $\hat{f}$ based on their orientations θ_p, Figure 2(c,d). We apply different filters perpendicular and parallel to the edges. As shown in Figure 3(e), a Laplacian is applied parallel to an edge to determine the edge's scale or length. Gaussian filters modeling the predicted positions of edge foci are applied on either side perpendicular to the edge. The responses of all edges are summed together to get the final filter response, Figure 2(e). As a result, edges that are equidistant and perpendicular to edge foci will reinforce each others' responses.

The filter applied parallel to an edge attempts to determine the scale of the edge, i.e. the linear length of the edge segment. A filter known for superior detection of scale [15, 9] is the Laplacian of Gaussian filter. As stated in [9], this filter will produce a maximal response at the correct scale without producing extraneous or false peaks. Our 1D filter is defined as:

$$u(x, \sigma) = -\sigma_u^2 \nabla^2 \mathcal{G}(x; \sigma_u), \quad (3)$$

where $\sigma_u = \sqrt{(\beta\sigma)^2 - (\alpha\sigma)^2}$ to account for the blurring already applied to the image. Scaling by a factor of σ_u^2 is required for true scale co-variance as shown by [9]. Varying the value of β will affect the size of the area around the edge foci that is reinforced by the individual edge responses, as shown in Figure 3(a,b,c). A value too large, Figure 3(a) will blur structural detail, while a value too small may suffer from aliasing artifacts and create multiple peaks if the edges are not exactly aligned perpendicular to the foci, Figure 3(c). We choose an intermediate value of $\beta = 0.5$ that is robust to noise, but does not overly blur detail, Figure 3(b).

The filter applied perpendicular to the edge allows edges of similar lengths to reinforce each others' responses at potential edge foci, as shown in Figure 1. We assume edge foci exist at a distance of σ perpendicular to the edge. As a result, our filter is the summation of two Gaussians centered at $-\sigma$ and σ:

$$v(x, \sigma) = \mathcal{G}(x - \sigma; \sigma_v) + \mathcal{G}(x + \sigma; \sigma_v). \quad (4)$$

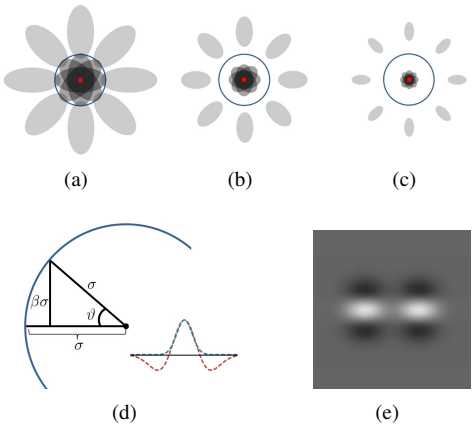

Figure 3. (a,b,c) illustration of different values of β for edges forming a circle. Grey ellipses represent the area of positive response for filter (e) applied perpendicular to the edges. (d) illustration of the computation of ϑ on a set of edges forming a circle, and overlapping Gaussian with Laplacian having twice the standard deviation. (e) the filter response g that is applied to normalized edge images.

The value of σ_v may be assigned based on the predicted variance of the edge foci. However, setting $\sigma_v = \sqrt{(\beta\sigma)^2 - (\alpha\sigma)^2}$ the same value as σ_u provides computational advantages. As we discuss in Section 3.2.1, the 2D filter resulting from convolving equations (3) and (4), shown in Figure 3(e), can be computed using steerable filters that are linearly separable.

For computational efficiency, we only evaluate the filters (3) and (4) at a discrete number of orientations θ_i, where $i \in N_\theta, (N_\theta = 8)$. For each orientation θ_i, we create an edge orientation image $\hat{f}_i$ that only contains normalized edges with orientations similar to θ_i. We softly assign edges to an edge orientation image $\hat{f}_i$ using:

$$\hat{f}_i(p) = \hat{f}(p)\mathcal{G}(\theta(p) - \theta_i; \vartheta), \quad (5)$$

where $\vartheta = \sin^{-1}(\beta)/2$, and $\theta_i = i\pi/N_\theta$. As illustrated in Figure 3(d), our Laplacian filter has a zero crossing at $\beta\sigma$, and we assume the edge focus point is a distance σ from the edge. For an object with locally constant non-zero curvature we can assign a value of $\sin^{-1}(\beta)/2$ to ϑ to match the widths of the Gaussian in equation (5) to the center of the Laplacian in (3), i.e. the standard deviation of the Gaussian is half the Laplacian, inset Figure 3(d).

As shown in Figure 2 and 3(e), if g_i is our 2D filter found by convolving our vertical filter (3) with our horizontal filter (4) and rotated by θ_i, we can compute our final response function h using:

$$h = \frac{1}{N_\theta} \sum_i h_i \quad (6)$$

where $h_i = \hat{f}_i \otimes g_i$.

3.2.1 Steerable filters

In this section, we describe how in practice we compute $h_i = \hat{f}_i \otimes g_i$ for all i. Naively convolving $\hat{f}_i$ with the 2D filter g_i is computationally expensive. Since filter (4) is the summation of two identical Gaussians, we can apply a single Gaussian blur and sum the result at two different offsets:

$$h_i(p) = \tilde{h}_i(p - p') + \tilde{h}_i(p + p') \tag{7}$$

where $p' = \{\sigma\cos(\theta_i), \sigma\sin(\theta_i)\}$ and $\tilde{h}_i = \hat{f}_i \otimes G_2^i$. G_2^i is the second derivative edge filter with orientation θ_i resulting from the convolution of a Gaussian and its second derivative. If $\sigma_u = \sigma_v$, G_2^i combined with (7) results in the same response as applying the filters (3) and (4). It is known [6] that G_2^i can be computed as a set of three 2D filters that are linearly separable:

$$G_{2a}(x, y) = \sigma_u^2 \mathcal{G}(y; \sigma_u)\nabla^2\mathcal{G}(x; \sigma_u) \tag{8}$$

$$G_{2b}(x, y) = \sigma_u^2 \nabla\mathcal{G}(x; \sigma_u)\nabla\mathcal{G}(y; \sigma_u) \tag{9}$$

$$G_{2c}(x, y) = \sigma_u^2 \mathcal{G}(x; \sigma_u)\nabla^2\mathcal{G}(y; \sigma_u). \tag{10}$$

Using equations (8), (9) and (10), we can compute $G_2^i = k_a(\theta_i)G_{2a} + k_b(\theta_i)G_{2b} + k_c(\theta_i)G_{2c}$ with:

$$\begin{array}{rcl} k_a(\theta_i) & = & -\cos^2(-\theta_i + \pi/2) \\ k_b(\theta_i) & = & 2\sin(\theta_i)\cos(-\theta_i + \pi/2) \\ k_c(\theta_i) & = & -\sin^2(-\theta_i + \pi/2) \end{array} . \tag{11}$$

In practice, G_2^i may also be computed by first rotating the image by $-\theta_i$, applying G_2^0 and rotating back. However, artifacts due to resampling may reduce the quality of the response. Computational requirements are similar for both techniques.

3.3. Scale space

In the previous section, we described how to compute a single filter response function h^k at the scale σ^k, where $k \in K$ is the set of computed scales. There are several methods for computing interest points across multiple scales depending on how image resampling is performed. For instance, a naive approach is to apply increasing values of σ to the original image to compute each scale. However at large values of σ, computing the filter responses can be expensive.

We use a popular approach that creates an octave image pyramid, in which the image size is constant in an octave of scale space and resized by half between octaves [11, 10]. An octave refers to a doubling in size of σ. This approach reduces artifacts resulting from resampling, while being computationally efficient due to the reduced image size at larger scales. Since we perform non-linear operations on our image, such as computing orientations and normalizing gradients, we are required to recompute $\hat{f}$ and θ to produce h^k at each scale k. Unlike our approach, methods based on linear filters such as the DoG interest point detector [11], may progressively blur filter responses for additional efficiency. Following [11] we compute three levels per octave, i.e. $\sigma^{k+1}/\sigma^k = 2^{1/3}$. Two additional padding scales are computed per level to aid in peak detection.

3.4. Maxima detection

Given a set of response functions h^k over scales $k \in K$ we want to find a set of unique and stable interest point detections. To accomplish this, we use the standard approach proposed in [11] for finding maxima in the responses spatially and across scales. A pixel p is said to be a peak if its response is higher than its neighbors in a $3 \times 3 \times 3$ neighborhood, i.e. its 9 neighbors in h^{k-1} and h^{k+1}, and its 8 neighbors in h^k. In addition to being a maxima, the response must also be higher than a threshold $\tau = 0.2$.

As first proposed by Brown and Lowe [3], we refine our interest point locations by fitting a 3D quadratic function to the local response h^k in the same $3 \times 3 \times 3$ neighborhood. The computed offset $\hat{x}$ from the original position x is found using:

$$\hat{x} = -\frac{\partial^2 h^{-1}}{\partial x^2}\frac{\partial h}{\partial x}. \tag{12}$$

4. Experimental Results

In this section, we show experimental results to illustrate the performance of edge foci interest points based on three different metrics: First, we provide an entropy measure to study the distribution both spatially and across scales of the interest points. Second, we score the interest points' repeatability, i.e. whether corresponding regions are chosen between images. Finally, we measure the uniqueness of the descriptors computed by the interest points to estimate the amount of ambiguity present during matching. We also evaluate the detectors on image alignment and retrieval tasks.

We compare the performance of our detector against some of the most commonly used detectors such as the Harris [7], Hessian [17], Harris/Hessian Laplace [17], MSER [13] and DoG [11] detectors[1] on a set of new datasets that capture non-linear illumination variations and changes in background clutter. Homographies are computed using ten or more hand labeled points to provide correspondences between pairs of images. We assume the scenes are either planar or taken at a far distance so correspondences can be well approximated by a homography. Figure 4 shows two example images from the 8 datasets used in this paper. The datasets Boat, Graffiti and Light were provided by and described in [17]. When required, the rotation of the interest points are computed using the method of [11].

Before we present quantitative experimental results, we show the response of our detector in comparison with the

[1]For the all the detectors except DoG we use the binaries from http://www.featurespace.org/, for DoG the binaries provided at http://www.cs.ubc.ca/~lowe/keypoints/ generate better results.

Figure 4. Example images from the 8 datasets used in the paper.

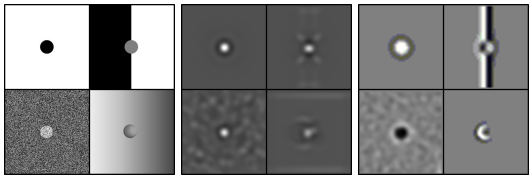

Figure 5. Filter response on four images (a) for the edge foci detector (b) and Laplacian detector (c).

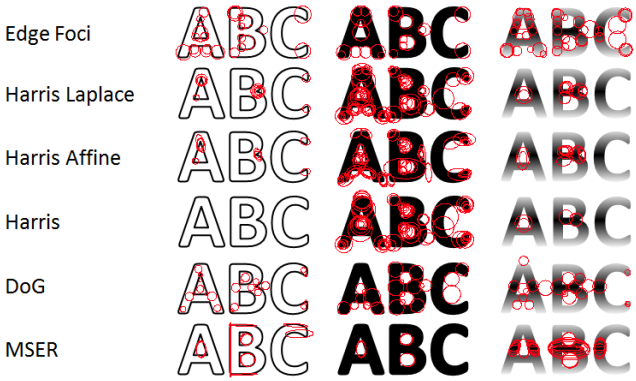

Figure 6. Detection results for different style fonts. Notice only Edge Foci produces repeatable interest points. Hessian detectors produce similar results to Harris.

Laplace detector[10] on a set of toy examples to provide additional insight. Figure 5 shows the filter responses of our detector and the Laplacian detector on a set of four examples for a fixed scale. Notice how the filter responses are more consistent for our edge based descriptor as compared to the intensity based Laplacian detector. The edge foci detector finds interest points at the correct scale and position on all four circles, where the Laplacian fails on the right two.

Since our detector is only dependent on normalized edge magnitudes, it is possible to find repeatable interest points on line drawings of images. This may be useful for detecting signs with different font styles as shown in Figure 6. Notice none of the other detectors produce repeatable interest points when the font changes style.

Entropy For many applications such as image matching and 3D reconstruction it is essential that local features detected in the image are well-distributed spatially and across scales. This is important for detecting objects occupying a small area of the image, and to remove the redundancy of overlapping interest points at neighboring scales.

In this section, we measure the distribution of the interest points in scale space based on their entropy. Intuitively, the positions of well distributed detectors should have a higher entropy than detectors with overlapping interest points. We compute entropy by discretizing positions and scales. For scales, we use the same discretization as was used to compute the detections. Spatially we discretize the image into bins of size $\xi\sigma^k$ where ξ is a scalar controlling the density of the bins. Since the spatial size of the bin is dependent on scale, there will be fewer bins at larger scales. That is, the size of the bins will have a ratio of $2^{1/3}$ between levels. We compute the contribution to each bin $b(x, y, k)$ using a Gaussian weighting on the positions of the detected interest points $m \in \mathcal{M}$:

$$b(p,k) = \frac{1}{Z} \sum_{m \in \mathcal{M}} \mathcal{G}(\|p - m_p\|/m_\sigma; \sigma_x)\mathcal{G}(k - m_k; \sigma_l) \tag{13}$$

where m_p is the position, m_σ is the scale and m_k is the scale level $(log(m_\sigma)/\log(2^{1/3}))$ of the interest point m, and Z is the normalization constant used to ensure all bins sum to 1. In our experiments we set $\sigma_x = 8$, $\sigma_l = 1$ and $\xi = \sigma/4$. These parameters result in overlapping contributions for detectors that extract overlapping image regions, assuming the descriptor has a size of $4m_\sigma$. A detector's entropy measure is then computed as:

$$\sum_p \sum_k -b(p,k) \log b(p,k) \tag{14}$$

Figure 7 summarizes the entropies for several detectors. We see that the edge foci interest points have higher entropy indicating that they have a better distribution across spatial locations and scales. Figure 8 shows an example of detected interest points and a visualization of the spatial distribution of $b(p, k)$ for the edge foci, DoG, Hessian Laplace and Harris detectors. We can clearly see that the edge foci interest points have a greater "spread" than the other interest point detectors that tend to cluster detections.

Repeatability The repeatability criterion measures the percentage of interest points that are detected at the same relative positions and scales across images. Considered in isolation, the repeatability score can be biased towards detectors that find overlapping interest points. That is, poor localization can be mitigated by the detection of redundant interest points. In practice this is undesirable for two reasons: First, redundant interest points require more storage, and increased computation for matching. Second, the descriptors corresponding to the interest points will not be

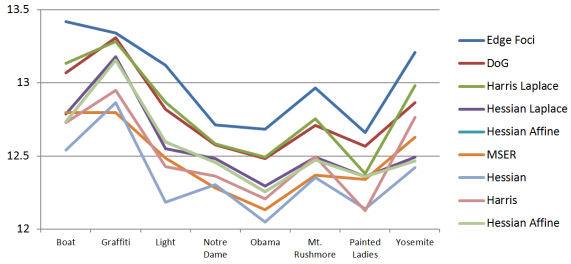

Figure 7. Entropy of interest point detectors across various datasets.

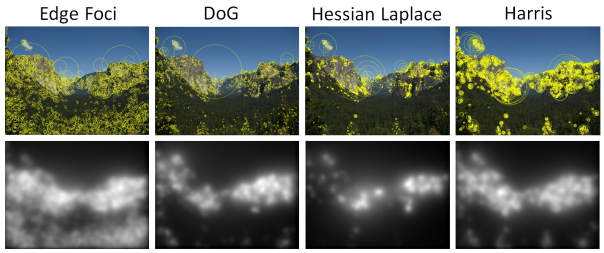

Figure 8. Visualization of the interest points and their spatial distributions for various detectors on Yosemite image.

unique, increasing the difficultly of matching and nearest neighbor techniques.

Two interest points, m and m', that are detected in different images are said to pass the traditional repeatability criterion [11, 16] if their relative positions and scales are within some scale normalized distance:

$$(\|m_p - m'_p\| - \varepsilon)/m_\sigma < \tau_p \tag{15}$$

$$\|m_k - m'_k\| < \tau_k. \tag{16}$$

where $\tau_p = 0.4$, $\tau_k = \log(1.3)$ and $\varepsilon = 2$ to account for small errors in the homography estimation. We normalize the distances by scale since the variance in the interest points' descriptors are related to scale normalized distances and not fixed distances. To ensure the consistent measurement of scale, the scales of all detectors were calibrated on a test image with various sizes of circles. If the projected interest points lie outside the other image, they are not used to compute the repeatability score.

We modify the traditional measure of repeatability to additionally penalize overlapping detections. Using the set of interest points $\mathcal{M}^*$ that passed the traditional repeatability criterion, we compute a distribution $B(p,k)$ using equation (13) similar to $b(p,k)$ used for our entropy measure. To only penalize detections resulting in descriptors that are more than half overlapping, we reduce the value of σ_x to 4. We assume a standard descriptor region size of 4 times the interest point scale. Our final measure of repeatability that encourages well distributed detections is:

$$\frac{1}{\#\mathcal{M}^*} \sum_p \sum_k \min\left(t, B(p,k)\right) \tag{17}$$

where $\#\mathcal{M}^*$ is the size of $\mathcal{M}^*$ and t is the product of the Gaussian normalization constants in Equation (13), $t = 1/(2\pi\sigma_x\sigma_k)$. Using Equation (17), if none of the interest points in $\mathcal{M}^*$ overlap, the repeatability score is the same as using a traditional repeatability criterion [11, 16]. However, if two interest points in $\mathcal{M}^*$ do overlap, their contribution is truncated by Equation (17) reducing the overall score.

For our experiments, we compare the repeatability of our Edge Foci interest points with the interest points generated from the Harris[7], Hessian[17], Harris/Hessian-Laplace[16, 17], MSER [13] and DoG detector [11] on the different datasets described in section 4. We tune each detector to generate approximately the same number of interest points per test image ranging from 700 to $2,500$ points. Figure 9 shows the repeatability measures for each dataset. Across the various datasets the edge foci detectors perform well, especially in those containing significant lighting variation (Inspiration point, Notre Dame, Mt. Rushmore, Painted Lady) and occlusion (Obama). MSER performs best on severe affine transformations (Graffiti). Hessian Laplace and Harris Laplace are generally within the top three performers, while Harris and Hessian typically perform worse due to poor scale localization. For comparison, the traditional repeatability scores averaged over the datasets without penalizing overlap [16] are EdgeFoci 24%, DOG 12%, HarLap 25%, Harris 17%, HesAff 18%, HesLap 22%, Hessian 19% and MSER 18%.

Descriptor uniqueness In the previous section, we measured performance based on repeatability of the detectors. While the repeatability measure describes how often we can match the same regions in two images, it does not measure the uniqueness of the interest point descriptors. Unique descriptors are essential for avoiding false matches, especially in applications with large databases. As stated before, redundant interest points across scales may create similar descriptors. Interest points that are only detected on very specific image features may also reduce the distinctiveness of descriptors.

We compute descriptor uniqueness using the state-of-the-art daisy descriptor [24] to describe the image regions for all detectors (similar results are achieved with SIFT [11].) Given the image descriptors for a pair of images, we find the nearest neighbors for each descriptor in the other image. Using our repeatability criterion described above, it is determined whether each pair of nearest neighbor detections correspond. Each pair is either assigned to a positive (repeatable) set of detections or a negative set. In the negative detections we also include the second best nearest neighbor match to obtain a better estimate of the distribution of negative matches. Finally, we create an ROC curve by varying the threshold on the descriptor distance ratio [11] and compute the number of false positives and true positives below the threshold. The distance ratio is the ratio between the distance of the best match and second best match in an

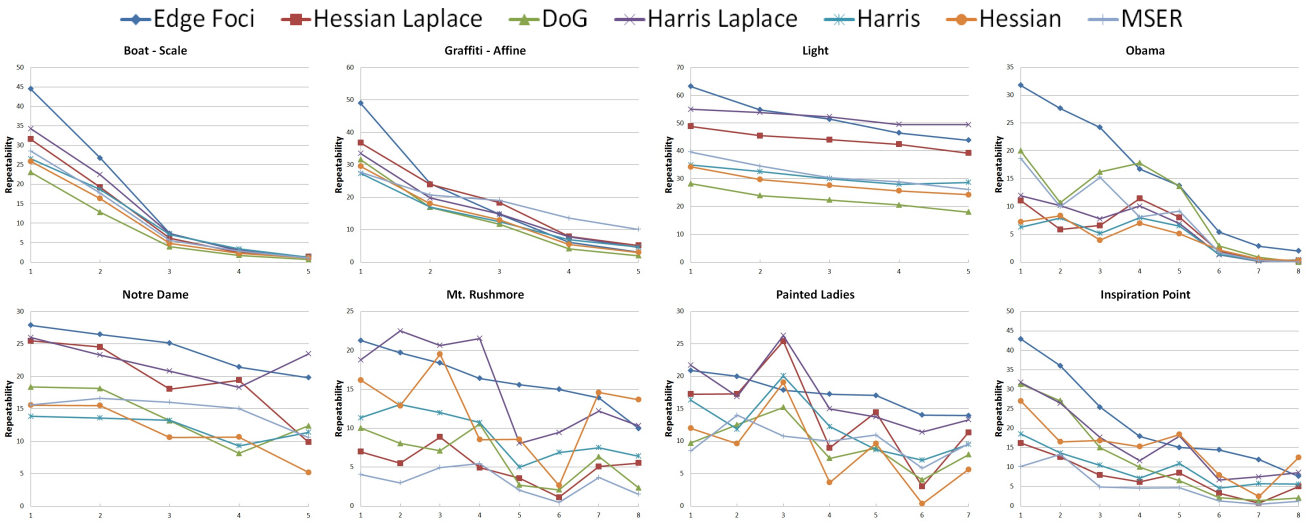

Figure 9. Graphs showing the repeatability with penalized overlap for 8 datasets using 7 different detectors.

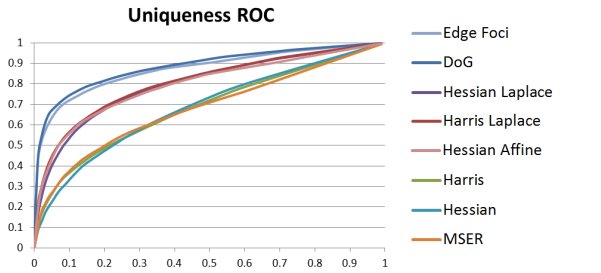

Figure 10. ROC curves showing the uniqueness of the descriptors generated from 8 different detectors averaged over 8 datasets.

image pair. We use the distance ratio or ratio test, since this criterion is commonly used to select matches and has shown good performance [11].

Figure 10 shows the uniqueness measure across all detectors as ROC curves. The plots show strong performance by both the edge foci and DoG detectors, while Harris, Hessian and MSER perform the worst.

Notice that the detectors have a tradeoff between uniqueness and repeatability. A detector that finds specific image structures may be very repeatable, but the corresponding features will not be unique. As a result, detectors need to be designed to perform well on both measures. For instance, Harris Laplace and Hessian Laplace are more repeatable but not as unique, where as DoG is more unique but not as repeatable. We believe, edge foci detectors maintain better balance between these tradeoffs.

4.1. Applications

In this section, we test the various approaches using applications that commonly use interest point detections. Specifically, we test the detectors in both image alignment and image retrieval applications.

We test the usefulness of the detectors in image alignment using a standard RANSAC approach. It is assumed

Table 1. (top) Percentage of image pairs in which RANSAC found the correct alignment for each descriptor type (EF: Edge Foci, DoG, HeL: Hessian Laplace, HaL: Harris Laplace, HeA: Hessian Affine, HaA: Harris Affine, Hes: Hessian, Har: Harris and MSER.) The mean of the Average Precisions (AP) for retrieving images of 11 buildings in the Oxford building dataset using bag-of-words (middle) and spatial verification (bottom).

Percentage correct: RANSAC alignment

EF	DoG	HeL	HaL	HeA	HaA	Hes	Har	MSER
53.0	40.9	47.4	50.9	47.3	48.8	38.4	39.6	43.9

Mean AP Bag-of-words: Oxford buildings

EF	DoG	HeL	HaL	HeA	HaA	Hes	Har	MSER
44.0	36.5	41.5	45.2	35.1	34.8	40.1	43.2	24.7

Mean AP Spatial Verification: Oxford buildings

EF	DoG	HeL	HaL	HeA	HaA	Hes	Har	MSER
47.0	43.8	41.8	47.2	37.0	38.2	37.7	39.3	30.5

the images are related by an homography. We evaluate their performance on the 8 image sets described above that contain varying amounts of illumination changes, scaling and projective distortions. The percentage of all images pairs from each dataset with correctly computed homographies are shown in Table 1. All of the of detectors perform well on the datasets Boat, Graffiti and Light from [17], but there exists more variation on the newer datasets. Overall Edge Foci (53.0%) performs the best followed by Harris Laplace (50.9%) and Harris Affine (48.9%).

Next, we test our detector on the image retrieval task using the Oxford Building dataset [20]. We use two approaches. In our first approach we use a Bag-of-words model with hierarchical K-means. Three levels are used with a branching factor of 80, resulting in 512,000 visual words. We use a stop list containing the 8,000 most commonly occurring words. A new vocabulary is built for each detector and matches are ranked based on the histogram intersection. Our second approach uses the same bag-of-

words model with the addition of spatial verification. Spatial verification is achieved using a three degree of freedom (position and scale) voting scheme between corresponding interest points. Using the code supplied by [20], we compute the mean of the average precision scores for each detector across all 11 buildings in the dataset. The results for both methods are shown in Table 1. Both Edge Foci and Harris Laplace achieve good results in both tests. It is interesting to notice that detectors with good localization (Edge Foci, DoG, MSER, etc.) get a larger performance boost from spatial verification than those with poor localization (Hessian, Harris).

Examining the performance of each query in isolation, it is clear that the accuracy of the detector is dependent on the content of the scene. While Harris Laplace and Edge Foci share similar average accuracies, the results on each query can vary significantly. For instance, Harris Laplace performs better on the "ashmolean" Oxford building, while Edge Foci performs significantly better on "bodleian" and "pitt rivers". For best results, it may be beneficial to use multiple detectors.

5. Discussion

Orientation dependent filtering is critical for localizing interest points using edge information. If a standard spatial filter such as a Laplacian is used directly on the edge responses [18], numerous false peaks and ridges occur. Due to the peakiness of our filter, we found it unnecessary to filter local maxima based on the ratio of principle curvatures as proposed by [11]. While not shown in this paper, the filter responses could be used for affine fitting similar to [16].

Due to the non-linear operations performed by our filter, and orientation dependent filtering, our detector is more computationally expensive than previous approaches [11, 13, 17]. The average run time for our detector on a 640×480 image is 4.25 seconds on a 2.53GHz Intel CPU. However, the filters applied to the images could easily be mapped to a GPU. The filters applied to the oriented normalized edge images $\hat{f}_i$ may also be computed in parallel.

In conclusion, we propose a method for detecting interest points using edge foci. The positions of the edge foci are computed using normalized edges that are more invariant to non-linear lighting changes and background clutter. We show improved results over previous works on numerous datasets with significant variation in lighting and occlusions.

References

[1] A. Baumberg. Reliable feature matching across widely separated views. In *IEEE Proc. of CVPR*, 2000. 1, 2

[2] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *IEEE Proc. of ECCV*. 2006. 2

[3] M. Brown and D. Lowe. Invariant Features from Interest Point Groups. In *BMVC*, 2002. 4

[4] J. Canny. A computational approach to edge detection. *IEEE Trans. on PAMI*, 8(6), 1986. 1, 2

[5] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. *IEEE Proc. of CVPR*, 2, 2003. 1

[6] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Trans. on PAMI*, 13, 1991. 4

[7] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. of the Alvey Vision Conference*, 1988. 1, 2, 4, 6

[8] T. Kadir, A. Zisserman, and M. Brady. An affine invariant salient region detector. In *IEEE Proc. of ECCV*. 2004. 1, 2

[9] T. Lindeberg. Scale-space theory: A basic tool for analysing structures at different scales. *Journal of Applied Statistics*, 21, 1994. 2, 3

[10] T. Lindeberg. Feature detection with automatic scale selection. *Int'l J. of Computer Vision*, 30, 1998. 1, 2, 4, 5

[11] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int'l J. of Computer Vision*, 60, 2004. 1, 2, 4, 5, 6, 7, 8

[12] M. R. B. M. Agrawal, K. Konolige. Censure: Center surround extremas for realtime feature detection and matching. In *IEEE Proc. of ECCV*, 2008. 2

[13] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10), 2004. 1, 2, 4, 6, 8

[14] K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *IEEE Proc. of ICCV*, 2001. 1

[15] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *IEEE Proc. of ECCV*, 2002. 1, 2, 3

[16] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *Int'l J. of Computer Vision*, 60, 2004. 1, 6, 8

[17] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Gool. A comparison of affine region detectors. *Int'l J. of Computer Vision*, 65, 2005. 1, 2, 4, 5, 6, 7, 8

[18] K. Mikolajczyk, A. Zisserman, and C. Schmid. Shape recognition with edge-based features. In *BMVC*, 2003. 1, 2, 8

[19] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. *IEEE Proc. of CVPR*, 2006. 1

[20] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *IEEE Proc. of CVPR*, 2007. 8

[21] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. In *IEEE Proc. of ECCV*. 2006. 2

[22] F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce. 3d object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. *Int'l J. of Computer Vision*, 66, 2006. 1

[23] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3d. In *SIGGRAPH*, 2006. 1

[24] S. Winder, G. Hua, and M. Brown. Picking the best daisy. *IEEE Proc. of CVPR*, 2009. 6

[25] C. L. Zitnick. Binary coherent edge descriptors. *IEEE Proc. of ECCV*, 2010. 1, 2