Robust Image Matching with Line Context

Wei Guan
wguan@usc.edu
Suya You
suyay@graphics.usc.edu

Computer Science Department
University of Southern California
Los Angeles, CA, USA

Abstract

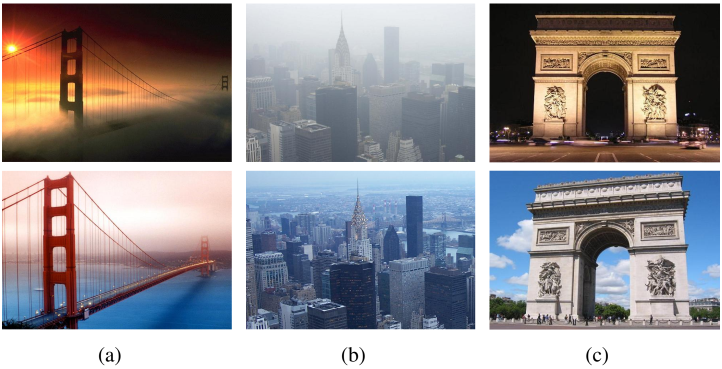
We present a novel type of feature that is robust to challenging images that are mainly due to large illumination variations. The keypoints are detected so that they are repeatable with consistent scales in images of different illuminations, and the descriptors are formed by the context of neighboring line segments. Similar to many popular local features, the proposed feature is robust to occlusions, viewpoint changes and image clutters. Differently, while most state-of-the-art features can only handle small or monotonic illumination changes, our proposed feature is capable to match images with large and non-monotonic illumination variations. Through extensive experiments, the proposed feature is demonstrated to be robust and efficient in matching such challenging images.

1 Introduction

Image spatio-temporal matching is a preliminary task in many computer vision systems that comprehend two or more images taken, for example, at different times, from different sensors, or from different aspects. In recent decade, many popular local features such as SIFT [14], SURF [3] and MSER [15] were proposed to match images. While these features are invariant in scales, rotations and robust to local deformations, they usually fail in cases when large illumination variations exist in the images, as the examples shown in Figure 1.

Matching images with large illumination variations is a fundamental and challenging problem. The matching is difficult because the intensities of an image are not linearly dependent on the lightings. Any changes in lighting will cause unpredictable changes in image intensities, which make most point-based features fail in the matching process. In this paper, we try to cope with this problem by proposing a new type of local feature, which describes the context information of neighboring line segments.

There are four challenges in constructing robust features that are based on line segments. The first challenge is to ensure that the interest points are repeatable with stable scales under large illumination changes. In our approach, we utilize edge pixels that have sufficiently large gradient values to make sure these pixels will probably appear in all images. Moreover, edge grouping is enforced so that same scales can be obtained on different images. The second challenge is to design a descriptor which is less sensitive to illumination variations but with sufficient distinctiveness. To form such a descriptor, the layout of edge pixels and signs of gradients rather than the gradient values (as in SIFT) are used. It is possible that the gradient signs may change in some rare cases. However, in most cases they remain the same because the signs are mainly determined by the reflecting characteristics which are



(a) (b) (c)

Figure 1: The challenging cases for image matching in different illuminating environments. The photos on top are taken in (a) the sunlight (b) a rainy day (c) night time. The photos at bottom are taken in a normal daytime.

invariant to illuminations. The third challenge is to maintain the feature robustness under unstable edge or segment detections. The unstability could be caused by occlusions, viewpoint changes or different sensors. We handle this by using multiple points representation of line segments. With multiple sample points, any part of segment will make partial contributions in computing the descriptors. The forth challenge is about fast matching and scalability to large datasets. There are algorithms that exploit complicated matching schemes to match two stereo images. However, these algorithms are designed to do pairwise matchings and not scalable to large databases. Since image matchings usually play an important role in retrieval techniques, it is more promising if the features can be applied in retrieval processes. To achieve scalable matchings, we need the feature to be in simple representation form so that tree structures can be used to reduce time complexity.

Given that our method is built on some of the previous work, we explicitly state our original contributions as follows:

1. We formally define a novel type of feature that is robust to challenging images that are mainly caused by large illumination variations. The feature is based on context information of line segments. The detailed keypoints detection and descriptor formation processes are presented in the paper.

2. We propose a scalable matching method based on the proposed features. The method can be applied to retrieval process so that images of the same scenes but under different illuminating environments can be successfully retrieved.

3. We have verified our proposed method on a self-constructed database which contains a large number image pairs under different illuminations.

2 Related Work and Motivation

In the past decade, a large number of local features are proposed in the computer vision community. These features are useful in many applications such as object recognition, image integration and image retrieval. They are necessarily to be distinctive, easy to extract, and robust to local deformations. According to the level of pixel groupings adopted in their descriptor construction process, these features can be classified into four categories. The first category is intensity based features. While we cannot list all of them, popular ones are

SIFT [14], SURF [3] and ASIFT [20] etc. These features make well use of the normalized gradients and histograms of gradient directions to form descriptors that are robust to local deformations and small illumination changes. Instead of using the gradient values, features in the second category make use of edge pixels or *edgels* that are generated by thresholding these gradients in a sophisticated way. There is a rich literature covering the concept of edge detection and we will not review them thoroughly here. Some popular edge-based features are [5, 16, 18, 21, 28]. The drawback of edgel based features is that they highly rely on edge detections and suffer from image clutters.

In recent years, more line based features were proposed [2, 7, 9, 24, 25]. While curve extractions are still challenges, lines are much simpler to detect and represent. A line is a higher-level grouping of edge pixels, and they are less sensitive to edge detections, clutters and illumination changes. However, most features in this category are designed for pairwise matching and not scalable to large dataset. The region based features in the forth category are quite different from each other. We still classify them into the same group because they all use within-region information in certain ways to form their descriptors. Some representative features and comparison studies are [6, 15, 19, 22].

Some features are specially designed to cope with complex illumination variations [8, 10, 11, 23, 26, 27]. These features belong to the first category and are based on intensity orders rather than values so that their descriptors are invariant to non-linear intensity changes. However, one limitation of such features is that they are only able to handle monotonic illumination changes. The feature proposed in this paper does not have such limitation.

Motivation. Through extensive experiments and based on existing research, we found that the best grouping level to handle large illumination variations is using curves or lines. Since curves require more sophisticated groupings and can be approximated by multiple line segments, we use line segments as the primitives of our feature descriptor. We call the proposed feature *Line Context*, since it is inspired by *Shape Context*. We will talk about the details of our proposed feature in the following sections.

3 Local Feature - Line Context

3.1 Interest Points Detection and Initial Scale

It is well known that edges are present at various scales. To detect edges at different scales we use multi-scale Canny edge detector [4] with Gaussian derivatives at several pre-selected scales. Adaptive Canny thresholds are used to ensure sufficient number of edges but our method is not sensitive to such threshold values. The scale step is set to 1.4. For each edge point, Laplacian operator is used to estimate the size of its neighborhood that has robust responses across all scales. Given the edge point, we compute Laplacian responses for several scales. Then we select the scale for which the response attains an extremum [14]. One property of Laplacian operator is that the scale that achieves extremum is equal to the distance to step-edges ([13]). After extremum detections, each edge point is assigned a scale σ.

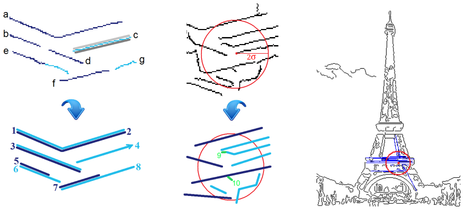
Not all detected edge points are considered keypoints. Two phases are used to remove unstable edges. The first phase is the above-mentioned Laplacian process. Those edges that do not attain a distinctive extremum over scales will be removed at the same time. The second phase is to remove those edges with homogeneous gradients, i.e. the points where the underlying curves have zero curvatures. We apply Harris matrix [12] to achieve this purpose. Let A be the Harris matrix calculated on the edge point and λ_1 and λ_2 be its two eigenvalues. The edge point will be rejected if $min(\lambda_1, \lambda_2) < T_{res}$, where T_{res} is the response threshold. Moreover, we can also control the number of interest points by threshold on Harris eigenvalues.

After the above two steps, the remaining edge points are relatively stable across different images and are considered keypoints. Each keypoint has a scale σ, which is calculated from the Laplacian operator. Due to intensity differences, the scales of same interest point may not be consistent on images with very different illuminations. So we consider σ as the *initial scale* of line context features. The grouping information will be enforced to obtain more accurate scales, which will be illustrated later.

3.2 Line Segments in the Context

The edge pixels are linked to connected curves at different scales. These curves are fitted by straight line segments with an approximation method similar to [1]. Each line segment is assigned an orientation. The orientation is the same as the line with direction determined by the sign of average gradients on its two sides along the segment. This is illustrated by *curve c* and *segment* 4 in Figure 2-(a).

As shown in Figure 2-(a), several cases need to be considered for representing curves with line segments. One curve may be fitted by multiple segments like *curve a*. Two segments with small gap in between are merged into one larger segment no matter they are on the same scale (*curve b* and *d*) or different scales (*curve f* and *g*). In the latter case, the merged segment only exists in the lower scale (*segment* 8). Besides, all the segments in higher levels are also segments (*segment* 1, 2 and 3) or part of segments (*segment* 5 and 7) in lower levels.



(a) (b) (c)

Figure 2: (a) Curves at various scales are fitted by line segments in different situations. Dark blue denotes higher scale level, and light blue denotes lower scale level. (b) Line segments within distance 2σ are considered context segments. (c) A keypoint and its context segments on an edge image.

For each keypoint, we need to find line segments in its neighborhood, which is also called *context* of the feature. The line segments lying inside or partially inside the context are called *context segments*. The initial scale σ provides an estimate size of searching area. However, since σ is equal to the distance to step edges, it is not surprising that the area within distance σ may not contain sufficient number of segments. By sufficiency, we mean that the area contains enough segments to be distinctive from another area. Through extensive experiments, we found that 2σ is the best trade-off between computations and distinctiveness. The technique of feature scale extension to involve more information is widely used and has been discussed in [17]. Let $d_{point-set}(v,S)$ be the shortest distance from point v to a set of points S. With keypoint k and all-segments $\{seg_i\}$, the segments set L in the context is defined as,

$$L(k) = \{seg_i | d_{point-set}(k, seg_i) \leq 2\sigma\} \tag{1}$$

Figure 2-(b) shows an example of context segments. All the segments in scale level σ and *lower* scales are included in the context as long as part of the segment is within distance 2σ. Note that segments with very small lengths are removed because these segments are unstable on different images (*segment* 9 and 10). The feature descriptor will be formed by the remaining segments. Figure 2-(c) shows a keypoint and its context segments on an image after edge detection.

3.3 Scale and Rotation Invariance

Feature scale. The *Line Context* feature is designed to be scale and rotation invariant. To achieve these two types of invariance, we need to define the exact scale and reference orientation for each feature point.

The scale 2σ that is used to search neighboring segments is not suitable to describe the size of keypoint context because many segments are partially outside this range. Therefore, we need to recalculate the scale for each feature point. Let m_i be the midpoint of segment seg_i in the context of keypoint k, $d_{point-point}(a,b)$ be distance between two points a and b. The scale s for the feature is defined as,

$$s(k) = \frac{\sum_{seg_i \in L(k)} d_{point-point}(k, m_i)}{|L(k)|} \tag{2}$$

During descriptor calculation process, all the context segments of keypoint k will be normalized by a factor of $s(k)$.

Feature orientations. Each keypoint descriptor is assigned a canonical orientation so that the descriptor is invariant to rotations. This orientation is determined by the dominant orientation of *context segments* in the *lowest* edge detection scale. A segment orientation histogram is created to vote for the dominance. The histogram has 36 bins covering 360 degree range of rotations. The vote from each segment is weighted by its length so that longer segments have larger contributions. The keypoint orientation is then determined by the peak in the histogram. To resolve ambiguities, those orientations that are very close to the peak value are also considered. In our experiments, we set the closeness ratio as 0.8. In other words, if the largest weight of voting is W_{max}, all those orientations with weights greater than $0.8W_{max}$ are considered new orientations. For each new orientation, a different descriptor is created. Therefore, each keypoint may have multiple dominant orientations and descriptors.

3.4 Line Context Descriptor

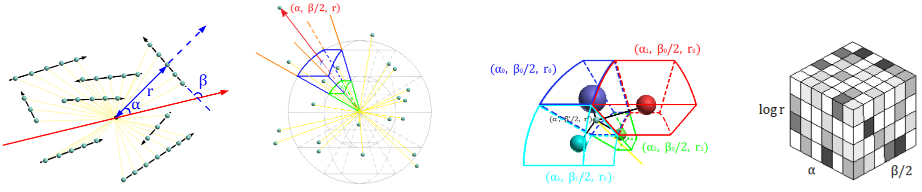
Multiple points representation. We know that the edge detection and linking are not always consistent, which means that many segments that are detected in one image may not be detected or partially detected in other images. Therefore, any particular single point on the segment is not sufficient to describe the whole segment. To resolve this problem, we use multiple sampled points as the representation of segments in the context.

To sample on the segments, we first normalize the context of keypoint k by its scale $s(k)$. After normalization, the average distance to the midpoint of these context segments would be unit 1. The lengths of these segments are also normalized accordingly. For each segment, starting from the midpoint along the segment itself in its two opposing directions, we sample the segment with step 0.1 unit. Figure 3-(a) shows an example of context segments sampling.

As shown in Figure 3-(a), four parameters are used to describe each sample point. They are 1) the distance r to the keypoint, 2) the angle $\alpha \in [0, 360)$ between the direction from keypoint to sample point and reference direction (keypoint dominant orientation), 3) the angle $\beta \in [-180, 180)$ between reference direction and the orientation of underlying segment and 4) the underlying segment scale σ. After sampling, all the sample points will be used to form the keypoint descriptor.

Log-2D-polar voting. Our goal is to produce a compact descriptor for each keypoint by computing a coarse histogram of the relative coordinates of segment sample points. We use a log-polar-like coordinate system to vote for the relative distances. Different from traditional log-polar coordinates, a 2D-polar coordinate is used since we have two angular parameters α and β. To be able to handle local deformations, we choose to use 6 equally spaced angle bins for both α and $\beta/2$, and 4 equally spaced log-radius bins for $\log r$. We use $\beta/2$ as the coordinates rather than β because $\beta/2$ is more convenient to be represented in the 3D sphere coordinates which is shown in Figure 3-(b).

The voting from each segment sample point is done as follows. Let p be the sample point, r', α', β' and σ be the four parameters belonging to p, σ_0 be the initial scale of the keypoint, which is also the highest scale for all context segments. Let $(\alpha_0, \beta_0/2, r_0)$ be the center point of the bin containing p. Besides the container bin, on each of the three coordinate directions, we also choose the neighbor bins that are closer to the sample point. This is illustrated in Figure 3-(c). Therefore, we have four bins $(\alpha_0, \beta_0/2, r_0)$, $(\alpha_1, \beta_0/2, r_0)$, $(\alpha_0, \beta_1/2, r_0)$ and $(\alpha_0, \beta_0/2, r_1)$, and each bin will be voted with certain weights. Their weights are denoted as W_0, W_α, W_β and W_r respectively.



(a) (b) (c) (d)

Figure 3: (a) multiple points representation of line segments (b) sample points in 3D sphere coordinates (c) voting to neighboring bins from a sample point (d) $\log r$, α, β histogram (darker grid represents larger weights)

Each sample point has a scale parameter σ. Those points with higher scales are more reliable than points in lower scales. Therefore, weights carried by sample points are set in proportion to their scales. Let sample point with the highest scale σ_0 carrying weight 1, so any point with scale σ will carry a weight σ/σ_0.

The weight voted to a bin by a sample point is determined by the weight carried by the point and the distance between the point and the bin center. If the bin contains the point, it is voted with the full weight of the point. In another word, $W_0(\sigma_0) = 1$ and $W_0(\sigma) = \sigma/\sigma_0$. For the other three bins, their weights are exponentially decreasing as the distance to their bin centers increase. The unweighted voting scheme could also be used, but the weighted voting has moderate improvements on robustness.

The accumulated weights from all sample points form a 3D descriptor which is shown in Figure 3-(d). Many bins have 0 vote which are similar to SIFT descriptors. Through experiments, we found that it is usually not necessary to cover segments in all scales. The scale σ_0 and one level lower are good estimations for most cases.

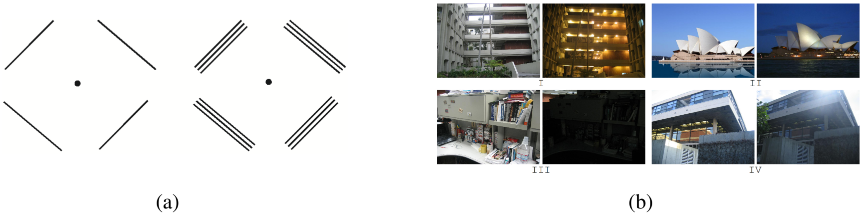
4 Matching Features and Images

Feature distance and similarity. Given two features, their distance is defined as the *1-norm* distance between the *unnormalized* descriptors. The feature has been normalized by its scale $s(k)$ before computing its descriptor. Once the descriptor is computed, it should not be normalized again because the descriptor represents the accumulation of weights rather than weight distribution. Figure 4-(a) shows an example of such a difference. As the figure shows, while left and right are two descriptors with different weights, they have the same weight distribution, and the descriptor will become identical after normalization.

Through experiments, we found that using 1-norm distance generates better performance than 2-norm distance. Let v_1 and v_2 be two descriptors, then their similarity $m(v_1, v_2)$ is defined as,

$$m(v_1, v_2) = 1 - \frac{||v_1 - v_2||_1}{||v_1||_1 + ||v_2||_1} \tag{3}$$

Matching images. To match two images, we find the most similar feature for each feature in the other image. The distance ratio method is used to remove ambiguous matches. In experiments, we set the ratio to 0.8. For image dataset, bag of features are used to retrieve relevant images. Since our feature is invariant to illumination changes, we are able to retrieve those images of the same scene but taken under different lightings.



(a) (b)

Figure 4: (a) Two features have different layouts of line segments. After normalization, their descriptors will be identical. (b) The example pairs in self-constructed dataset.

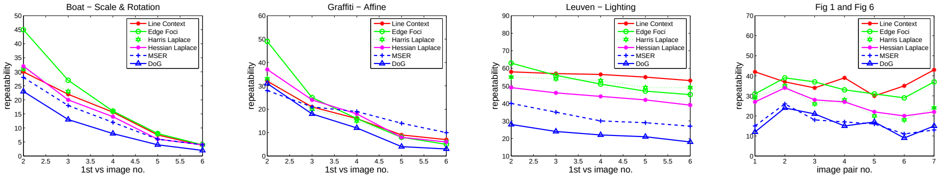
5 Results and Discussions

In this section, we show experimental results to demonstrate effectiveness of the proposed *Line Context* feature. We will evaluate the performance of both its detector and descriptor.

To validate our detector/descriptor, we have collected images under various conditions from both the Internet and taken by ourselves (as in Figure 4-(b)). These images have large illumination variations as well as scale and viewpoint changes. The ground truth correspondences are obtained by enforcing both homographic (larger error tolerance due to non-planar parts) as well as epipolar (smaller error tolerance) constraints. We compare our proposed feature with other local features on standard dataset as well as constructed dataset.

Repeatability. Repeatability is the percentage of keypoints that are detected at same positions and scales across two or more images. For our experiments, we compare the repeatability of Line Context detector with other detectors such as Edge Foci [28], Harris/Hessian Laplace [19], MSER [15] and DoG detector [14]. We tune each detector to generate approximately the same number of interest points per image ranging from 800 to 2000 points. Figure 5 shows the repeatability measures for Mikolajczyk's dataset [17] and constructed dataset. In the figure, (a) and (b) show results of handling scale, rotation and affine transformations, while (c) and (d) are results for illumination variations. As the results show, Line Context detector is comparable with Hessian Laplace for handling scales, rotations and affine transformations, in which cases Edge Foci performs the best. However, for images with illumination variations, our detector is competitive with Edge Foci and slightly better in cases when the variations are large.

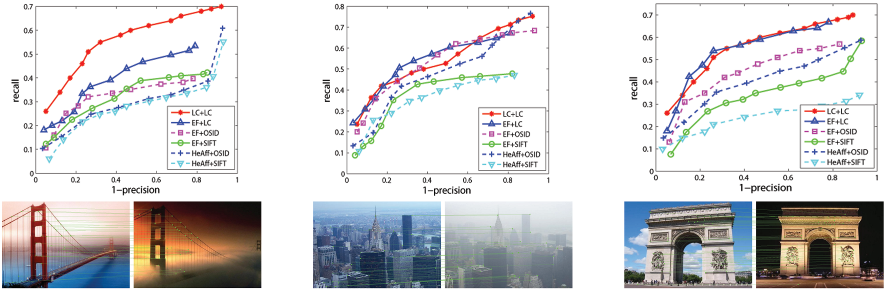
Feature matching. For the same detector, the matching performance varies by using different descriptors. We compare our Line Context descriptor with OSID [23], which is one of representative descriptors that are specially designed to handle illumination variations. Different combinations of detectors (LC - Line Context, EF - Edge Foci, HeAff - Hessian Affine) and descriptors (Line Context, OSID, SIFT) are compared.



(a) (b) (c) (d)

Figure 5: Repeatability test on both standard dataset (Mikolajczyk's, (a) - scale and rotation, (b) - affine, (c) - lighting) and collected dataset ((d) - lighting).

Figure 6 shows point correspondences obtained by matching Line Context features (detector+descriptor) for images in Figure 1. We define precision as the percentage of correct matches out of all matches and recall as percentage of keypoints with correct correspondences. The comparisons with other detector/descriptor pairs are shown above each image pair. It can be observed that images in pair (a) have very different tones, and images in pair (c) are almost in two opposite illuminating conditions, the combination of our detector and descriptor significantly outperforms other combinations on these challenging cases. For image pair (b), which is taken with blurry effects, our detector+descriptor has similar performance to Edge Foci combined with OSID and outperforms the others.

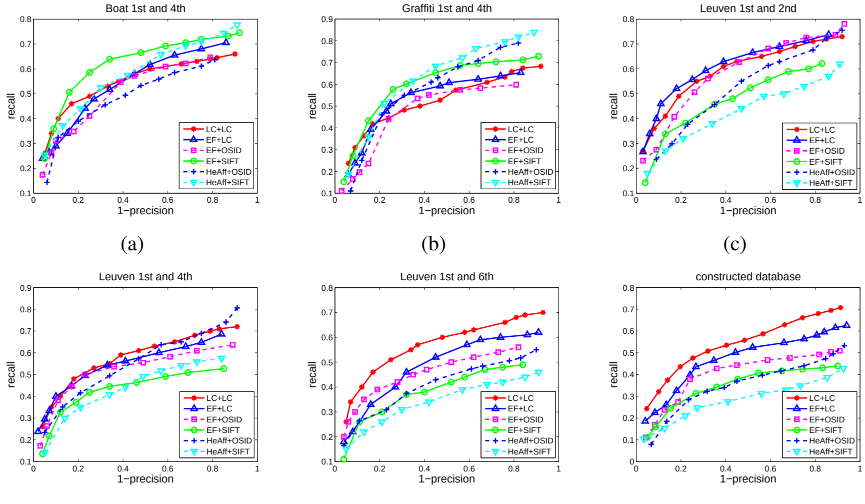


(a) (b) (c)

Figure 6: Point correspondences obtained by matching Line Context features and comparison results with other local features for images in Figure 1.

We have tested our feature on more than 120 pairs of challenging images that are taken under different circumstances. The challenges are mainly caused by various illumination changes, such as different lighting sources, day-or-night illuminating and different weather conditions etc. The average precision-recall curve is drawn in Figure 7-(f). It clearly shows that our feature performs better than the other features on these challenging cases.

We have also tested our approach on Mikolajczyk's dataset which is one of the standard datasets to verify different feature characteristics. In their illumination cases, the images are less challenging compared to ours in two aspects. The viewpoint changes are smaller and the illumination changes are relatively uniform and monotonic. Figure 7-(c), (d) and (e) show the comparisons on handling these cases. It is demonstrated that as the illumination differences become larger, our detector and descriptor combination is more robust compared to others. For "Leuven" 1st and 6th images, our feature performs much better than the others. We also test Line Context on scale/rotation and viewpoint changes, and the results are reported in Figure 7-(a) and (b) respectively. The proposed feature is comparable to others in handling scale/rotaion variances. For images with viewpoint changes, our feature is not as good as some top performers such as SIFT with Hessian Affine detector. This is because under viewpoint changes, the parameters r, α and β for each sample will be affected at the same time, which cause variances on the voting results. However, the performance on viewpoint changes is close to many classic descriptors and is quite acceptable in many applications.



(a) (b) (c)

(d) (e) (f)

Figure 7: Precision-recall measures on Mikolajczyk's standard dataset (a)-(e) and constructed dataset (f). The cases handled are (a) scale+rotation (b) viewpoint changes (c)-(e) illumination changes (f) more challenging illumination changes.

Running performance. In Table 1, we show run-time comparisons with several popular features (detector+descriptor). For MSER detected regions, SIFT descriptors are used. As can be observed, our approach can detect more correct matches than most of these features. The Line Context extraction and matching time are similar to others. While in some examples LS can detect more correct matches, its matching process takes much longer due to its complicated matching scheme. For the same reason, LS is not suitable for tree structures without major modifications thus not scalable to a large number of images.

It should also be noted that one limitation of Line Context feature is that it does not perform well in images with all small or fine-grained textures. In these images, there are no explicit curves or lines. The textures are made up of tiny segments in all different directions. When the context size is small, the voting is more sensitive to changes or noises. In this case, we cannot form stable context segments around keypoints, and the computed descriptors are not robust.

	1-(a)	1-(b)	1-(c)	4(b)-I	4(b)-II	4(b)-III	4(b)-IV	Extract	1-Pair	n images
SIFT	2/11	6/13	1/17	2/21	2/15	1/14	5/21	0.9-1.3s	0.1-0.2s	$O(\lg n)$
SURF	0/7	4/9	0/12	0/11	1/8	0/12	4/17	0.3-0.4s	0.1-0.2s	$O(\lg n)$
SC	3/17	3/11	1/13	2/16	3/18	1/12	3/18	0.7-1.2s	0.2-0.5s	$O(\lg n)$
LS	29/37	51/55	19/21	23/35	11/15	72/77	52/65	3-4s	12-18s	$O(n)$
MSER	0/2	1/7	2/9	2/7	0/5	0/8	1/12	0.2-0.3s	0.2-0.4s	$O(\lg n)$
Line Context	31/47	36/53	49/60	36/53	32/41	42/58	34/39	0.8-1.5s	0.2-0.3s	$O(\lg n)$

Table 1: Comparison of different features (SIFT, SURF, Shape Context, Line Signature, MSER+SIFT, Line Context) on images in Figure 1 and Figure 4-(b). The number of correct matches over the number of detected matches is reported. It also compares feature extraction time, matching time for one pair of images, and complexity for matching with n images. Tree structures can be applied for some features to achieve $O(\lg n)$ complexity.

6 Conclusion

We have presented a new type of feature, called *Line Context*. The feature is formed by describing the layout of line segments in the context of an interest point. Pixel grouping is enforced after Laplacian-of-Gaussian to detect a more repeatable scale under large illumination changes. Moreover, to achieve robustness to unstable edge detections and linkings, multiple sample points are used to represent each segment. A voting scheme is then proposed to generate a three dimensional histogram. The unnormalized weights in the histogram form a line context descriptor. Similar to most local features, line contexts are robust to occlusions and viewpoint changes. Significant improvements are achieved on matching challenging images such as large illumination variations.

References

[1] N. Ansari and E. J. Delp. On detecting dominant points. *Pattern Recognition*, 24(5): 441–451, 1991.

[2] H. Bay, V. Ferrari, and L. Van Gool. Wide-baseline stereo matching with line segments. In *CVPR*, 2005.

[3] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *ECCV*, pages 404–417, 2006.

[4] J. Canny. A computational approach to edge detection. *PAMI*, 8:679–698, 1986.

[5] O. Carmichael and M. Hebert. Shape-based recognition of wiry objects. In *CVPR*, pages 401–408, 2003.

[6] C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein, and J. Malik. Blobworld: A system for region-based image indexing and retrieval. *Visual Information and Information Systems*, 1999.

[7] P. David and D. DeMenthon. Object recognition in high clutter images using line features. In *ICCV*, 2005.

[8] B. Fan, F. Wu, and Z. Hu. Aggregating gradient distributions into intensity orders: A novel local image descriptor. In *CVPR*, 2011.

[9] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid. Groups of adjacent contour segments for object detection. *PAMI*, 30(1):36–51, 2008.

[10] R. Gupta and A. Mittal. Smd: A locally stable monotonic change invariant feature descriptor. In *ECCV*, 2008.

[11] R. Gupta, H. Patak, and A. Mittal. Robust order-based methods for feature description. In *CVPR*, 2010.

[12] C. Harris and M. Stephens. A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference*, pages 147–151, 1988.

[13] T. Lindeberg. Scale-space theory in computer visionobject recognition by affine invariant matching. *Kluwer Academic Publishers*, 1994.

[14] D. G. Lowe. Object recognition from local scale-invariant features. In *ICCV*, pages 1150–1157, 1999.

[15] J. Matas, O. Chum, M. Urba, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *BMVC*, pages 384–396, 2002.

[16] J. Meltzer and S. Soatto. Edge descriptors for robust wide-baseline correspondence. In *CVPR*, pages 1–8, 2008.

[17] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *PAMI*, 2005.

[18] K. Mikolajczyk, A. Zisserman, and C. Schmid. Shape recognition with edge-based features. In *BMVC*, pages 779–788, 2003.

[19] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. In *IJCV*, 2005.

[20] J.M. Morel and G. Yu. Asift: A new framework for fully affine invariant image comparison. In *SIAM Journal on Imaging Sciences*, pages 438–469, 2009.

[21] J. Malik S. Belongie and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 24(4):509–522, 2002.

[22] E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *CVPR*, pages 1–8, 2007.

[23] F. Tang, S. Hwan Lim, N.L. Chang, and H. Tao. A novel feature descriptor invariant to complex brightness changes. In *CVPR*, pages 2631–2638, 2009.

[24] L. Wang, U. Neumann, and S. You. Wide-baseline image matching using line signatures. In *ICCV*, 2009.

[25] Z. Wang, F. Wu, and Z. Hu. Msld: A robust descriptor for line matching. In *PR*, 2009.

[26] Z. Wang, B. Fan, and F. Wu. Local intensity order pattern for feature description. In *ICCV*, 2011.

[27] Y. Yu, K. Huang, W. Chen, and T. Tan. A novel algorithm for view and illumination invariant image matching. *IEEE Transactions on Image Processing*, 21(1):229–240, 2012.

[28] C. Lawrence Zitnick and K. Ramnath. Edge foci interest points. In *ICCV*, 2011.