

# Illumination and Affine- Invariant Point Matching using an Ordinal Approach

Raj Gupta

Indian Institute Of Technology Madras  
I.I.T Post Office Chennai-600 036

rgupta@cse.iitm.ernet.in

<http://www.cs.iitm.ernet.in/~rgupta/>

Anurag Mittal

Indian Institute Of Technology Madras  
I.I.T Post Office Chennai-600 036

amittal@cse.iitm.ernet.in

<http://www.cs.iitm.ernet.in/~amittal/>

## Abstract

*We present an approach for illumination and affine-invariant point matching using ordinal features. Ordinal measures for matching only consider the order between pixels and not the absolute intensity values which enables them to be invariant to a monotonic change in intensities. The utility of such measures has been demonstrated in the past for point matching for some applications such as background subtraction and stereo matching. However, invariance of such methods to geometric transformations has been limited leading to their inapplicability for more generic matching applications such as object recognition, wide-baseline stereo matching, mosaicing or tracking points on moving objects. In this paper, we extend such methods for use in such applications. The method is invariant to an affine transformation in the patch, which makes it applicable to a variety of applications. At the same time, our method is robust to different types of noise processes possible in a real scene. Experiments indicate favorable performance when compared to other state-of-the-art methods for affine-invariant point matching.*

## 1. Introduction

Point matching is an important task in many Computer Vision applications such as object recognition, stereo matching, mosaicing and auto-calibration. Many methods have been proposed in the past for this task. Such methods range from simple methods like normalized cross-correlation to more advanced techniques such as SIFT which is based on the use of gradient histograms. The important criteria for a match measure are its robustness to changes in illumination and the different types of perturbations / noise processes possible in the patch while maintaining a high matching performance. Another important criteria for such match measures, particularly in applications where the viewpoint changes, is its robustness to geometric

transformations that occur naturally in images taken from different viewpoints (for e.g. in wide baseline matching).

In this paper, we present an approach that uses ordinal measures in order to obtain robustness to illumination changes. At the same time, our method is invariant to an affine transformation in the image patch around a feature point so that it is useful in applications where there are viewpoint changes. Ordinal measures are useful in obtaining invariance to illumination changes since they only consider the relative order of the pixels in a patch and not the absolute intensity values. Thus, they can handle a non-linear change in intensity, the only requirement being that the changes be monotonic. There are non-linear illumination changes due to camera characteristics such as non-linear response function, gamma correction, defocus, saturation and under-saturation, or other non-linear effect include storage/transmission artifacts such as image compression. Thus, a model that is more generic than the normally used linear change model can provide more robustness to illumination changes in many applications.

Based on this idea, several point matching algorithms have been developed. The census transform [25] looks at all the neighbors of a given pixel and creates a vector from the order of this pixel with respect to the neighbors. Image matching can then be performed by correlation in this transformed space. Bhat and Nayar[1] improved upon such measure by a carefully designed distance between two rank permutations. Mittal and Ramesh [15] improved upon these methods by combining intensity and rank information so that the cost of a change of order between pixels is proportionate to the intensity difference between them. This makes the method relatively stable to Gaussian noise. We use some of the ideas from this paper in order to make our feature point matching invariant to a monotonic change in intensity.

While such methods are useful in many applications, they cannot handle a large change in the viewpoint and hence are inapplicable for many tasks such as wide-baseline matching[18, 24, 4], point matching across cameras, object

recognition [10], texture recognition [9], image retrieval [11, 19], robot localization, video data mining[20] and recognition of object categories. In order to make such an approach applicable to these applications, one has to make the matching invariant to the geometric transformation taking place in the patch around a point. In this paper, we develop approaches to efficiently search for an affine transformation that aligns one patch with another such that the intensity transformation between the two is monotonic. Such an approach results in an illumination and affine-invariant point matching algorithm that can be used for many applications where one needs to match points that might be observed by two cameras under different lighting conditions and from different viewpoints.

### 1.1. Related Work in Affine Invariant Matching

There has been quite a bit of work in affine invariant point matching[14]. The idea that is often used is to detect points with sufficient texture contrast and to determine a transformation of the region around the point so as to bring the region to a normalized shape. Certain features are then extracted from these normalized shapes. These feature vectors are then compared in order to determine a score for the matching. Techniques for detection and normalization of such affine “covariant” regions include the Harris-affine detector[12, 18], the hessian-affine detector [12], maximally stable extremal region detector[6], edge-based and intensity extrema-based detectors[24] and entropy-based detectors[22]. Techniques for feature vector descriptors and matching include SIFT[10], shape context[16], steerable filters[2], PCA-SIFT[7], spin images[17] and moment invariants[8]. SIFT has been shown to outperform other feature descriptors[13]. Other methods for obtaining affine invariance include image lines connecting interest points [5, 23] and invariant vertical line segments[21].

Almost all of these methods assume a linear change in intensity in order to perform matching and even to obtain the geometric normalization for matching. We show that our method is able to handle illumination changes much better than most methods and is also able to handle other image artifacts such as image defocus and image compression much better. The reason is that often these factors introduce a non-linear change in intensities and hence lead to an error in these methods while our method is relatively unaffected by such changes.

The paper is organized as follows. Section 2 describes the extraction of distinctive feature points that can be used for matching. Section 3 describes the affine and illumination-invariant feature point matching technique that is used. Finally, section 4 presents the experimental results and comparison to previous techniques for generic point matching.

## 2. Feature Point Extraction

The first step in our approach is the detection of keypoints in an image. Keypoints are highly distinctive points so that they can be easily distinguished from other similarly extracted points in the same or another image. Also, the same point must be detected as a keypoint even when the image undergoes some changes due to changes in the illumination or viewpoint or due to some noise processes such as Gaussian noise, image blur or compression. Towards this goal we propose the following solution.

### 2.1. Detection of Discriminative Feature Points

Points having a high contrast or “texture” around them are generally distinctive since typically, there is a very low probability that the same texture pattern is observed in another image patch not arising from the same or similar object. The complexity of the texture patterns that we can reliably distinguish from each other is dependent on the flexibility of the image transformations that we allow. For instance, since a projective transformation allows more distortion in an image patch as compared to an affine transformation, we will need the matching pattern to be more complex in order to determine reliably if two patterns can be said to be occurring due to the same object (since simple patterns will match with many other patterns, not giving us discriminability). In order to handle transformations upto affine, which are generally sufficiently flexible for matching, we will take the following approach.

First, we note that the intersection point of lines (i.e. corner points) remains unchanged in an affine transformation. Such corner points have been used quite popularly for robust and distinctive feature point extraction [3]. Here, we try to find corner points that are at the intersection of two or more lines with the added constraint that matching along these lines is sufficient to determine the affine transformation between the patches. Two lines matches with scale information are sufficient to determine the affine transformation (since the center is common, there are only four unknowns in the affine transformation and each line with scale information provides two constraints). Similarly, three lines intersecting at a point with only one of them having scale information are also sufficient. However, four “non-scale” lines are insufficient to determine the affine transformation since the constraint from the last line is not independent of the constraint from the first three lines.

Thus, in our method, we will try to find linear gradient structures (lines) that intersect at a center reference point  $P$  being tested. To test for a line, we consider two lines in the shape of a cone passing through  $P$  on either side of the test line direction (Fig. 1). We have used a cone angle of  $10^\circ$  and lines that are about 20 pixels long in our experiments. The lines extend in both directions since lines in opposite

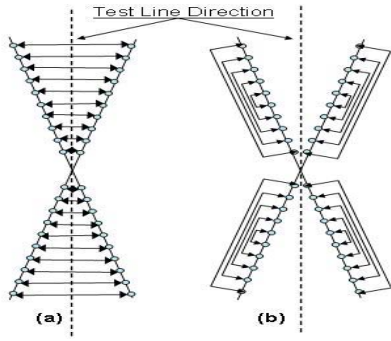


Figure 1. Point Matching along a particular direction. The first image shows the points that are compared “across” the direction to test for a line along this directions the second shows the points that are compared “along” the line to test for a contrast along this direction ( to obtain scale information ).

directions yield the same constraint in the determination of the affine transformation. Along these lines, we take the differences across the supposed line as shown in Figure 1 (a) and sum such differences to obtain a value for the ‘evidence’ of a gradient across this line. This is done for some 20 directions around the point. Then, we determine (circular) local maximas of such evidence to determine directions with strong evidence of linear gradient structures.

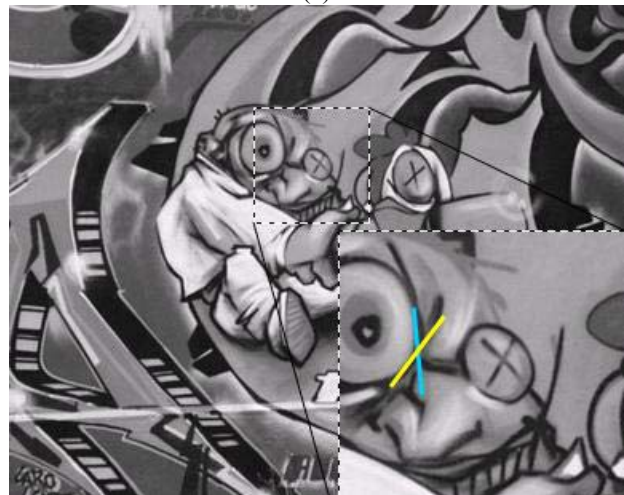
Now, if there is some contrast along a line, then one can determine the scale at which this line may match with another similar line in another patch. This will give us an additional constraint for determining the affine transformation. For this purpose, we calculate the distance measure value along a line as shown in Fig. 1 (b). If one is able to obtain a high value for the contrast along the line, then we will determine and store the scale along this line at which the calculated differences are the maximum. This “max” scale can later be compared to the scale of another line while matching. If the scale information is available for two lines, then these two lines, when matched, will yield the affine transformation. However, if scale is available for only one line, then we will need two other lines for matching. Thus, a keypoint is detected if either there are two lines with scale information or three lines with one line having scale information.

Fig. 2 shows the result of keypoint detection on two images that have a viewpoint change between them. As shown, the keypoints are detected in the two images at the same point, which is the intersection of two lines with scale information.

A last constraint we put on the keypoints is that they must have a higher contrast value compared to their neighborhood. The idea is that a high contrast region will typically lead to a lot of ‘hits’ around it and we would like to extract the point with the best “hit” among all such points. For com-



(a)



(b)

Figure 2. The figures show a keypoint that was extracted in the two images of a scene with different viewpoints. The keypoint extractor extracts the same point in the two images as the intersection of two lines remains the same under affine transformation. If we are able to determine the scale also along these lines, then we can compute the full affine transformation between the patches.

paring the neighborhood points for maximum contrast, the sum of the contrast values from all of the local maxima lines is used. The keypoint detection described above is done at several scales in order to handle possible scale changes.

## 2.2. Creating Feature Vectors

Given distinctive keypoints in an image, we would like to store some characteristics of such points so that they can be matched later on with other points. Such stored information is typically called a feature vector in related work [10]. In our method, the feature vector consists of the sampled points along the different directional regions which were



Figure 4. The figure shows the matching of one of the keypoints that was extracted in two images taken from different viewpoints. The matched lines with the scale factors determine the whole affine transformation.

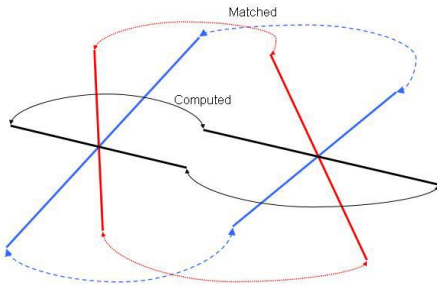


Figure 3. The figure shows the idea behind line matching. If we are able to establish correspondence between the red and blue lines along with the scaling factor, then the correspondence and scaling factor for the rest of the lines can be determined.

used for texture contrast computation in the previous section. These sampled points will be used for illumination invariant matching as will be described in the next section.

### 3. Feature Point Matching

Given the feature vectors as defined in the previous section, we wish to match the feature vectors as obtained from two different points such that the matching is invariant to an affine transformation. Furthermore, we would like the matching to be invariant to a monotonic change in intensity. Towards this end, we propose the following approach. Our first step will be to find an illumination invariant measure for matching two lines.

### 3.1. Illumination-Invariant Line Matching

Let us assume in this section that we are given two lines which need to be matched and also the scaling factor at which these lines are to be matched is given. Then, we consider certain point pairs “across” the line (Fig. 1 (a)). Since the scale factor is known, one can determine the corresponding point pair on the other line. Given such a point pair, we try to determine if the order of the pixels in this point pair has flipped. The idea behind this test is that, in the case of a monotonic change in intensity, the order of the pixels remains the same. A penalty is there to be ascribed to an order change. However, since noise can also easily cause this change if the two pixels are very close to each other in intensity, we will give a penalty to an order change equal to the lower of the difference in the intensity values of the two pixels being compared. Since either of the two lines being compared could cause the flip, the lower of the such differences is considered. The summation of the penalties over the whole line is taken as the difference between the two lines:

$$d(l^1, l^2) = \sum_{i \text{ s.t. } d_i^1 * d_i^2 < 0} \min(|d_i^1|, |d_i^2|)$$

where

$$d_i^1 = I_{i1}^1 - I_{i2}^1, \quad d_i^2 = I_{i1}^2 - I_{i2}^2,$$

Here,  $I_{i1}^1$  and  $I_{i2}^1$  are the intensities of the two points in the  $i$ -th pair on the first line ( $l^1$ ) and  $I_{i1}^2$  and  $I_{i2}^2$  are the intensities of the corresponding points on the second line ( $l^2$ ). A similar test is performed for point pairs “along” the line (Fig. 1 (b)). The two difference values are summed to get the final difference value between the two lines.

Given such a method for line matching, we develop an approach for affine-invariant point matching in the next section.

### 3.2. Affine-Invariant Feature Point Matching

Let  $F_1$  and  $F_2$  be two feature vectors that need to be compared. In the first phase, we match the local maximas only and try to determine the affine transformation between the two feature vectors. A local maxima line is matched using the approach given in the last section. The scale factor between the two lines is computed using the “max” scale value which was computed during feature point extraction.

We try to match either two local maxima lines with scale information or three lines with one line having scale information. If we are able to get such a match, then we compute the affine transformation determined by these line matches. Now, using this affine transformation, one can determine the correspondence between the lines of the two feature points and also the scale at which such lines should be matched (Fig. 3). The final match score for all feature points is taken as the sum of the match scores of the different lines. We may note here that if certain lines have very low contrast across or along them, then they are automatically given lower weight according to our difference measure and thus, only the high contrast lines are in significant for matching.

Fig. 4 shows an example of a successful match between two keypoints, where two local maxima lines with scale information was sufficient to find the whole affine transformation, which was then used for matching all the other lines. A thorough evaluation of our approach compared to other methods is presented in the next section.

### 4. Results and Comparisons

We compared the performance of our matching method with several other methods for affine-invariant region matching using a standard dataset[14] of images that contains changes in rotation, scale, viewpoint, illumination, blur and compression. The review paper[14] gives the results of six affine-invariant region detectors [12, 18, 6, 21, 22, 14] combined with the SIFT[10] operator for computing and matching the feature descriptor. The region detectors detect certain special regions and transform them into a normalized region. A feature descriptor is then used to compute a feature vector which is then matched with the feature vector of a second region. The SIFT feature descriptor which uses a gradient histogram has been found to be the most robust in several studies[13] and hence this has been used in this study to evaluate the different region detectors.

Some images from the database of the review paper that were used for testing are shown in Fig. 5. The results from our method were then compared with the other methods on these images. Fig. 6 shows the results of keypoint detection. The left plots show the repeatability scores, which is the percentage of keypoints that are common in the two images, while the right plots show the total number of detected keypoints. Our repeatability scores were mostly higher than

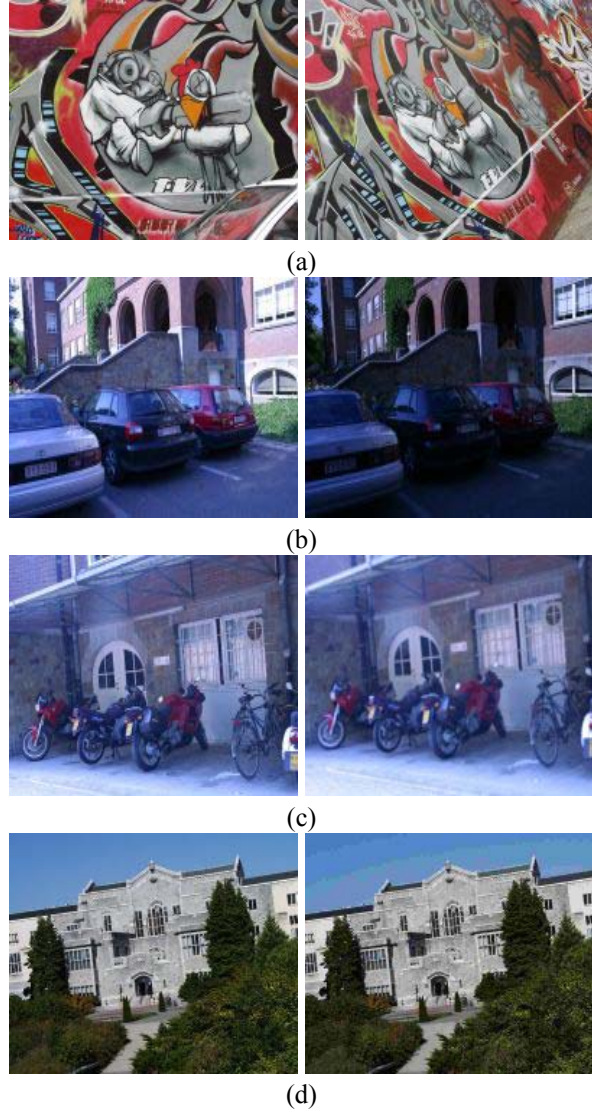


Figure 5. Example of images taken from the Database used in the paper. The two images have (a) Viewpoint Change (b) Illumination Change (c) Image blur, and (d) JPEG compression.

other methods, although the total number of keypoints extracted was somewhat low. Fig. 7 shows the results of matching the keypoints extracted, as compared to other techniques. Again, we get very high matching scores, but fewer number of matched points. The low number of keypoints may be due to our criteria for a point to be classified as a keypoint which necessitates atleast two lines *with* scale information or three lines with one scale information. This condition may not be easily satisfied, especially in natural scenes that may not have too many straight lines. However, for most applications, the number of points found is probably sufficient, especially given the accuracy of the method that may reduce the need for outlier rejection techniques.

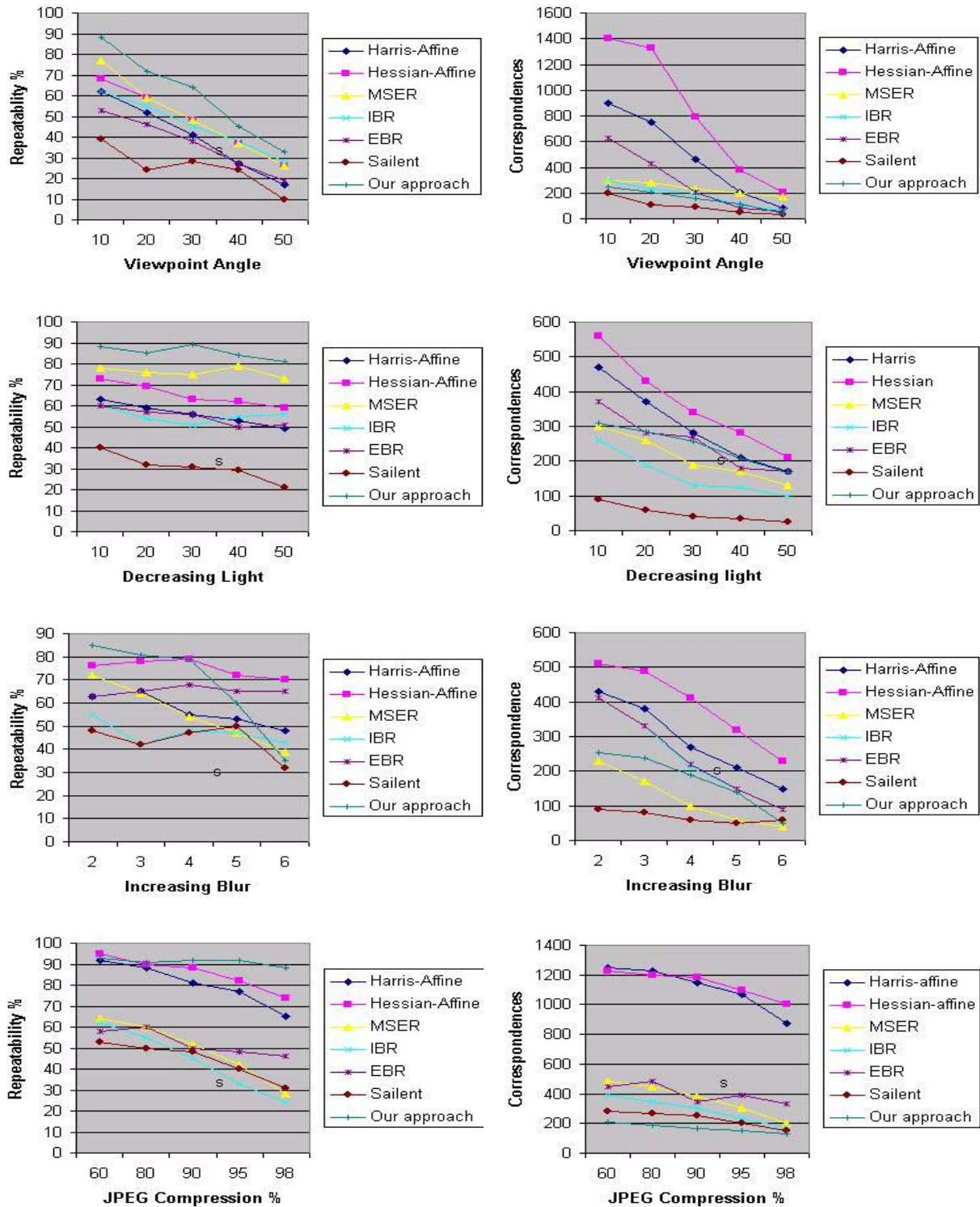


Figure 6. Results of keypoint extraction on the images shown in Fig. 5. The left plots show the percentage of keypoints that were common in the two images compared to the total number of keypoints extracted while the plots on the right show the total number of such common keypoints found.

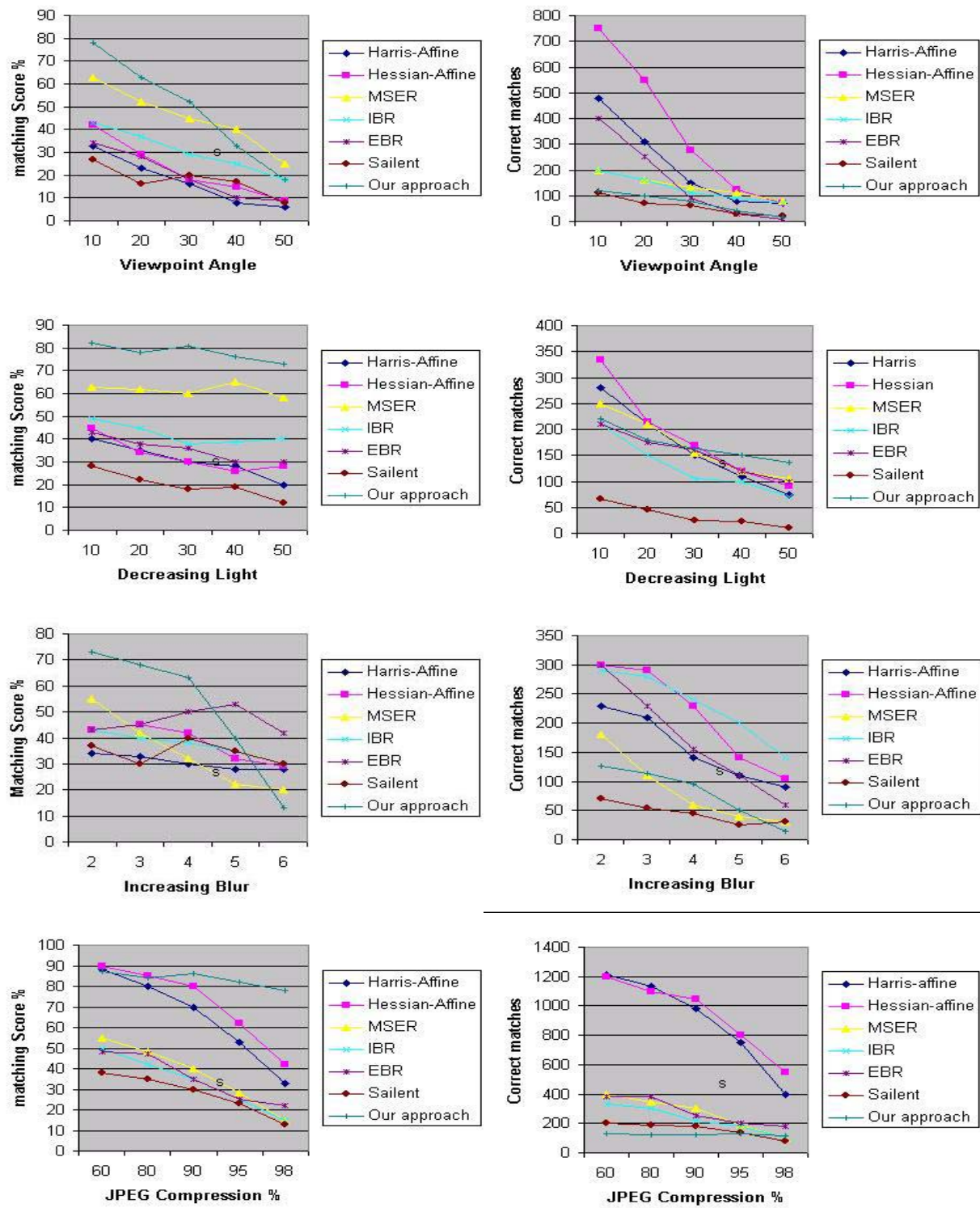


Figure 7. Results of keypoint matching on the images shown in Fig. 5. The left plots show the percentage of the matches that were correct, while the right plots show the total number of correct matches found.

Our method was found to be quite robust to changes in the illumination, as expected, and very little performance degradation was observed, even with a 50% reduction in the light. The method was also quite robust to JPEG compression and image blur for reasonable levels of blur. The reason is that such artifacts have very low probability of causing a flip, especially across high intensity gradients. On the other hand, such artifacts introduce non-linear changes in the intensity, which is a problem for most methods. For very high blurs, the image edges are quite blurred out and we are not able to detect the keypoint accurately. This is a problem only when the images are taken with a very bad focus, which is typically not the case. Our method was also found to be quite robust to a viewpoint change, although no method performed well when the angle of change was too large ( $> 30$  degrees).

## 5. Conclusion

In this paper, we have proposed a novel approach for interest point detection and matching. The keypoint detection was found to be quite robust to image rotation, scale and an affine transformation. The matching method is invariant to a monotonic change in intensity and is robust in the presence of different types of noise, such as Gaussian noise and Salt and Pepper noise, and also image artifacts such as image compression and blur. Experiments were conducted on a standard dataset and show that the performance of our approach was better than existing approaches in most cases, especially when there was a significant degree of illumination changes or other non-linear artifacts such as image blur and compression. The generic point matching algorithm presented in this paper can be used for a variety of applications such as mosaicing, image retrieval, stereo matching, object recognition, person tracking and identification, and robot localization.

## References

- [1] D. N. Bhat and S. K. Nayar. Ordinal measures for image correspondence. *IEEE Transactions on PAMI*, 20(4):415–423, 1998. 1
- [2] W. Freeman and E. Adelson. Shape matching and object recognition using shape contexts. *IEEE Trans. PAMI*, 13(9):891–906, 1991. 2
- [3] C. Harris and M. Stephens. A combined corner and edge detector. *In Alvey Vision Conference*, pages 147–151, 1988. 2
- [4] H. Bay, V. Ferraris, and L. Gool. Wide-baseline stereo matching with line segments. *CVPR*, 1:329–336, 2005. 1
- [5] J. B. J. Matas and J. Kittler. Object recognition using the invariant pixel-set signature. *In Proceedings of the British Machine Vision Conference, London, UK*, pages 606–615, 2000. 2
- [6] M. U. Jiri Matas, Ondrej Chum and T. Pajdla). Robust wide baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, pages 761–767, 2004. 2, 5
- [7] Y. Ke and R. Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. *CVPR*, 2004. 2
- [8] T. M. L. Van Gool and D. Ungureanu. Affine/photometric invariants for planar intensity patterns. *ECCV*, pages 642–651, 1996. 2
- [9] S. Lazebnik, C. Schmid, and J. Ponce. Sparse texture representation using affine-invariant neighborhoods. *CVPR*, pages 319–324, 2003. 2
- [10] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004. 2, 3, 5
- [11] K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. *IJCV*, pages 525–531, 2001. 2
- [12] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *IJCV*, 1(60):63–86, 2004. 2, 5
- [13] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1615–1630, 2005. 2, 5
- [14] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. *IJCV*, 65(1):43–72, 2005. 2, 5
- [15] A. Mittal and V. Ramesh. An intensity augmented ordinal measure for visual correspondence. *CVPR*, pages 849–856, 2006. 1
- [16] J. M. S. Belongie and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. PAMI*, pages 509–522, 2002. 2
- [17] C. S. S. Lazebnik and J. Ponce. Sparse texture representation using affine-invariant neighborhoods. *CVPR*, pages 319–324, 2003. 2
- [18] F. Schaffalitzky and A. Zisserman. Multi-view matching for unordered image sets. *Proc. Seventh European Conf. Computer Vision*, pages 414–431, 2002. 1, 2, 5
- [19] C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *IEEE Trans. PAMI*, 19(5):530–534, 1997. 2
- [20] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. *ICCV*, pages 1470–1478, 2003. 2
- [21] T. T. T. Goedeme and L. V. Gool. Fast wide baseline matching for visual navigation. *CVPR*, pages 24–29, 2004. 2, 5
- [22] A. Z. T. Kadir and M. Brady). An affine invariant salient region detector. *In the Proceedings of the 8th European Conference on Computer Vision*, pages 345–357, 2004. 2, 5
- [23] D. Tell and S. Carlsson. Combining appearance and topology for wide baseline matching. *ECCV*, pages 68–81, 2002. 2
- [24] T. Tuytelaars and L. V. Gool. Matching widely separated views based on affine invariant regions. *IJCV*, 1(59):61–85, 2004. 1, 2
- [25] R. Zabih and J. Woodfill. Non-parametric local transforms for computing visual correspondence. *In ECCV*, pages 151–158, 1994. 1