A Descriptor based on Intensity Binning for Image Matching

B. Balasanjeevi and C. Chandra Sekhar
Indian Institute of Technology, Madras, India

Keywords: Image Descriptor, Computer Vision, Image Matching.

Abstract: This paper proposes a method for extracting image descriptors using intensity binning. It is based on the fact that, when the intensities of the interest regions are quantized, the pixels retain their bin labels under common image deformations, up to a certain degree of perturbation. Consequently, the spatial configuration and the shape of the connected regions of pixels belonging to each bin become resilient to noise, which, as a whole, capture the topography of the intensity map pertaining to that region. We examine the effect of classical image deformations on this representation and seek to find a compact yet robust representation which remains unperturbed in the presence of noise and image deformations. We use Oxford dataset in our experiments and the results show that the proposed descriptor gives a better performance than the existing methods for matching two images under common image deformations.

1 INTRODUCTION

Local features have gained widespread attention recently, due to their robustness to image deformations, image occlusion and changes in viewpoint. A local feature encodes the intrinsic pattern that captures the essence of a region of interest, independent of other such regions. It does not necessarily correspond to any meaningful part of the scene and hence can be selected, although not exclusively, based on the underlying image properties such as intensity, texture and color. Local features have been used successfully for image matching (Tuytelaars and Van Gool, 2004), object recognition (Viola and Jones, 2004; Leibe et al., 2008; Fergus et al., 2003; Lowe, 2004; Nister and Stewenius, 2006; Zhang et al., 2007; Berg et al., 2005), texture recognition (Zhang et al., 2007), image retrieval (Mikolajczyk and Schmid, 2001), building panoramas (Brown and Lowe, 2003) and social media (Snavely et al., 2008; Kennedy and Naaman, 2008; Agarwal et al., 2009).

The process of constructing local features involves two stages. The first step (localization) consists of detection of Interest Regions (IR) which possess high information content while being robust to the image deformations like blur, illumination changes, scaling, rotation and affine transformations. Various methods for detecting interest regions were reviewed in (Tuytelaars and Mikolajczyk, 2008), of which Hessian-Affine detector (Mikolajczyk and Schmid, 2005) was shown to perform better than other methods.

The next step involves building a descriptor for each interest region obtained from the previous step, such that the representation is compact, discriminative, generalized and robust to image deformations and noise.

There is a rich set of existing methods for extracting descriptors which are presented in Section 2. The proposed method is discussed in Section 3 and the results are presented in Section 4.

2 METHODS FOR EXTRACTION OF DESCRIPTORS

Many techniques have been developed for describing the interest regions. One of earliest known works is steerable filters (Freeman and Adelson, 1991) which steer derivatives in a particular direction making them invariant to rotation. Johnson et al. (Johnson and Hebert, 1997) introduced a representation called spin image. Baumberg (Baumberg, 2000) proposed a descriptor which uses a multi-scale Harris feature detector (Harris and Stephens, 1988), with a set of invariants robust to local linear transformations forming the descriptor. Berg, et al. (Berg et al., 2005) proposed a deformable shape matching framework, which incorporates geometric blur descriptor (Berg and Malik, 2001) as well as the geometric distortion between pairs of corresponding points. Lowe (Lowe, 2004) proposed a Scale Invariant Feature Transform

Balasanjeevi B. and Chandra Sekhar C..
A Descriptor based on Intensity Binning for Image Matching.
DOI: 10.5220/0004827700960103
In *Proceedings of the 3rd International Conference on Pattern Recognition Applications and Methods* (ICPRAM-2014), pages 96-103
ISBN: 978-989-758-018-5

(SIFT), which combines a scale invariant region detector and a descriptor based on the gradient distribution in the detected regions. Ke and Suthankar (Ke and Sukthankar, 2004) developed the PCA-SIFT descriptor, which represents local appearance by principal components of the normalized gradient field, which is more compact than the standard SIFT representation. Mikolajczyk and Schmid (Mikolajczyk and Schmid, 2005) proposed GLOH (Gradient Location and Orientation Histogram), in which they modified SIFT by using a circular gradient location orientation grid, as well as the quantization parameters of the histograms. Heikkil et al. (Heikkil et al., 2009) proposed CS-LBP (Center Symmetric-Local Binary Patterns), a variant of LBP, which is more compact than LBP and is computationally less expensive than LBP and SIFT. Lazebnik (Lazebnik et al., 2005) extracted a sparse set of affine regions and then constructed descriptors based on spin images and SIFT descriptor. Bay, et al. (Bay et al., 2008) proposed an efficient implementation of SIFT by applying the integral image to compute image derivatives. Chen, et al. (Chen et al., 2010) proposed a descriptor inspired by Weber's law, which constitutes two components: one based on relative pixel differences and other on the orientation of the pixel, which were used to construct the descriptor. Cheng, et al. (Cheng et al., 2008) introduced a local image descriptor robust to general image deformations by using multiple support regions of different sizes centered on the interest point. Winder and Brown (Brown et al., 2011) proposed a framework for combining various descriptors and learned an optimal parameter setting to maximize the matching performance. Many of the aforementioned descriptors were evaluated in (Mikolajczyk et al., 2005) and that SIFT and GLOH were found to perform better when compared to other descriptors under common image deformations. In this paper, we will be using these two descriptors along with CS-LBP for benchmarking the performance of the proposed descriptor.

3 THE PROPOSED METHOD

3.1 Motivation

An image descriptor should mirror the topography of the underlying intensity map such that the representation is distinctive, compact and most importantly robust to common image deformations.

To construct a descriptor which satisfies these properties, we first analyze how the intensity map is affected by deformations such as blur, contrast changes, and similarity and affine transformations. Instead of considering the raw pixel intensities, we quantize the intensity map and analyze the effect of each deformation on the bin labels of the pixels as follows:

1. Consider an image pair in which one of the images is perturbed with one of the aforementioned deformations.
2. Extract the IRs from the image pair and find the corresponding regions using the method discussed below (Section 3.1.1).
3. Quantize the intensities of the corresponding region pairs between the images and compute the proportion of the pixels which retain the bin label.

3.1.1 Identifying the Corresponding Regions

The IRs are extracted from each image using the Hessian-Affine detector, which outputs elliptical regions. Given an IR (r_j) in one image, its corresponding IR (CR) in the other image is obtained by projecting the region under a homography $\mathcal{H}$ relating the images, and finding the IR in the second image with the highest overlap. Let $\mathcal{R}^{(i)}$ represent the set of IRs in image *i*. For any region $r_j \in \mathcal{R}^{(1)}$, let r'_j represent its projection under $\mathcal{H}$. Then the corresponding region of r_j is given as,

$$\mathrm{CR}(r_j \in \mathcal{R}^{(1)}) = \{r^*_j \in \mathcal{R}^{(2)} \,|\, \forall r_2 \in \mathcal{R}^{(2)}, \\ overlap(r'_j, r_2) \leq overlap(r'_j, r^*_j))\} \quad (1)$$

The overlap between two elliptical regions r_1 and r_2 is computed based on the method proposed in (Mikolajczyk and Schmid, 2005). The minimal rectangular region bounding the ellipse pair is sampled and the amount of overlap is computed as the ratio of the number of points belonging to both the regions to that belonging to either of the regions. That is,

$$overlap(r_1, r_2) = \frac{r_1 \cap r_2}{r_1 \cup r_2} \quad (2)$$

Here, the numerator denotes the area of the intersection and the denominator denotes the area of the union of the elliptical regions.

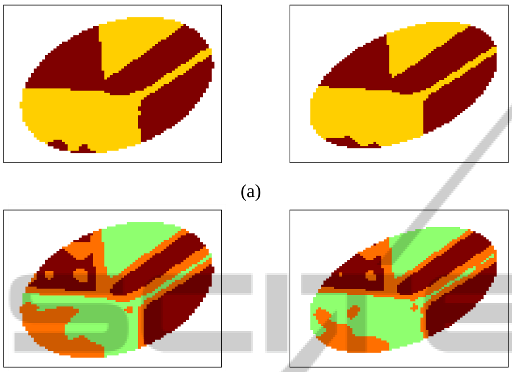
3.1.2 Observations

The evaluation of the above method was done on Oxford data set [1], which contains image sets for benchmarking the descriptors under a variety of image deformations. Each set consists of 6 images, with the

[1] Oxford data set is available at http://www.robots.ox.ac.uk/∼vgg/research/affine.

reference as image 1, and the images 2 to 6 being the increasingly perturbed versions of the reference image.

Figure 1 shows the quantized regions of a pair of corresponding IRs obtained using the procedure mentioned above under blur. It is evident from the figure that quantizing the intensity maps makes the IRs strikingly similar.



(a)

(b)

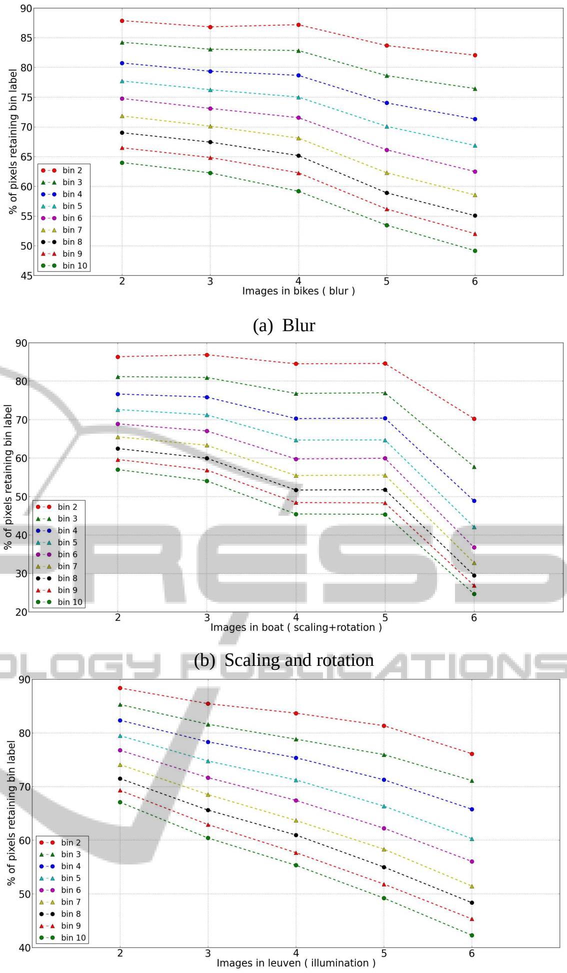
Figure 1: The quantized intensity maps of a pair of corresponding regions using (a) 2 bins (b) 3 bins.

Figure 2 shows the percentage of pixels across all IRs which retain their bin labels when the intensity range of each IR is quantized into n bins, where n is varied from 2 to 10, under a variety of image deformations. As is evident from the figure, this quantity decreases with increasing perturbation and also with increasing number of bins, due to the fact that the noise resilience of a pixel's bin label varies inversely with the width of the bin. That is, a slight perturbation is sufficient to effect a change in the bin label as the number of bins increases. It should be noted that, almost all of the pixels (90%) retain their bin labels when $n = 2$. Thus, we can claim that,

The bin label of a pixel is more or less invariant under classical image deformations, up to a certain degree of perturbation.

Now, if the bin labels of pixels remain unchanged, then the connected regions of the pixels belonging to a bin will retain their shapes and spatial configuration. That is,

If we consider a pair of corresponding regions, the shapes and the spatial configurations of the connected regions of pixels belonging to the same bins will be similar.



(a) Blur

(b) Scaling and rotation

(c) Illumination changes

Figure 2: Evaluation of the effect of different image deformations on the bin labels for different types of distortions (a) blur (b) scaling and rotation (c) illumination changes.

3.2 Constructing the Descriptor

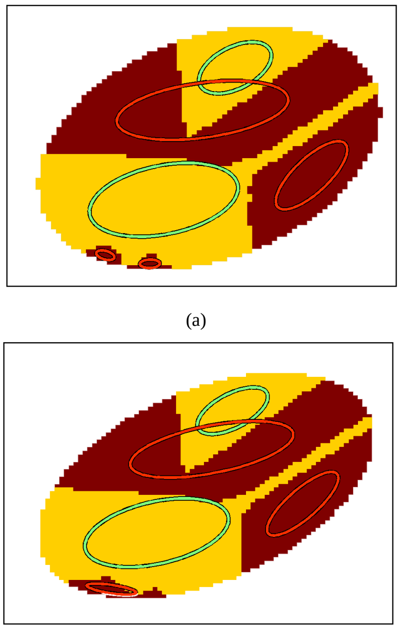
Thus, based on this observation, we can assert that the quantized map captures the topography of the intensity map of an IR. But such a representation cannot be used as such due to its high dimensionality. To obtain a compact representation, we compute second order central moments which capture the shape and spatial configurations of the connected regions, since the center of mass and the spread of the region are captured more succinctly. Also, such a construction has an additional advantage of being a generalized description of the intensity map, making it more resilient to image perturbations.

The descriptors are constructed as follows. Given an image, the IRs are first extracted using Hessian-Affine detector and each IR is quantized using one of the three methods discussed in Section 3.5, using n bins. We call such a quantized map as an n-map. To represent an n-map, we compute the second order central moment, i.e., we fit an ellipse, to every connected region of pixels belonging to each bin.

Algorithm 1: Compute Descriptors.

```
procedure $\mathcal{D}$ = COMPUTEDESC($r$,$n$)    ▷ interest region $r$ with $n$ bins
    $\mathcal{B} \leftarrow$ Bin($r$,$n$)    ▷ Quantize $r$ using $n$ bins
    $\mathcal{D} \leftarrow [\,]$    ▷ Initialize descriptor
    for $bin$=1 ... $n$ do
        $K_{bin} \leftarrow [\,]$    ▷ Initialize $K_{bin}$
        $n_{bin} \leftarrow$ #connected regions in current bin
        for $reg$=1 ... $n_{bin}$ do
            $\mathcal{E}_{reg} \leftarrow$ FitEllipse($\mathcal{B}$,$bin$,$reg$)
            $K_{bin} \leftarrow [K_{bin}\ \mathcal{E}_{reg}]$    ▷ Append ellipse
        end for
        $\mathcal{D} \leftarrow [\mathcal{D} K_{bin}]$    ▷ Append $K_{bin}$
    end for
end procedure
```

The steps involved in constructing the descriptor are shown in Algorithm 1. The function *FitEllipse* fits an ellipse to a connected region of pixels, by computing its second order central moment. An ellipse $\mathcal{E}$ is represented by the 5-tuple, (u,v,a,b,c) satisfying the equation $a(x-u)^2+2b(x-u)(y-v)+c(y-v)^2=1$. Figure 3 shows the fitted ellipses for the 2-maps of a pair of corresponding regions.


(a)

(b)

Figure 3: (a) and (b) show the ellipses fitted for the connected regions of each bin of the 2-maps of a corresponding regions pair. The green ellipses belong to bin 1 and the red ones to bin 2.

The descriptor obtained using this method is a variable length encoding of an IR, consisting of a set of ellipses. The descriptor is of the form $\mathcal{D}$:= $\{K_1,K_2,\ldots,K_n\}$ where n denotes the number of bins used for quantization; K_i:$\{\mathcal{E}_j\}_{j=1}^{|K_i|}$ denotes the set of ellipses fitted for each connected region of pixels belonging to the i^{th} bin; $\mathcal{E}_j$ being the ellipse fitted to the j^{th} connected region.

3.3 A Better Representation of the Regions

The descriptor presented in Algorithm 1 fits a single ellipse to a connected region. But, if the region is not elliptical, say, if the region is in the form of a ring, the ellipse thus fitted does not accurately capture the shape of a region, making the descriptor less discriminative and impair the performance. To circumvent this drawback, we fit multiple ellipses to a connected region of pixels so that the ellipses fit the region more ”tightly”. That is, we try to minimize the number of pixels which lie outside the fitted ellipses and the empty area of the fitted ellipses. This is equivalent to the geometric set covering problem, which can be stated as,

Given a grid, a set of points which are required to be covered and a set of forbidden points, we need to reduce the number of ellipses which cover all the required points and none of the forbidden points.

To this end, we define the error of the fit ε as

$$\varepsilon = \frac{A_{ell}}{N_{cov}} \qquad (3)$$

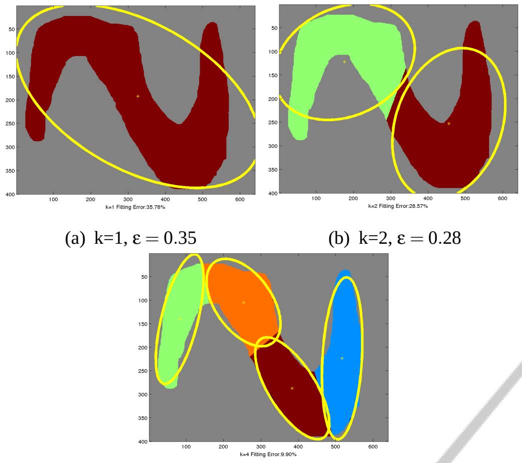
where N_{cov} represents the number of points in the region that are covered by the ellipses and A_{ell} represents the total area of the ellipses. That is, for the error of the fit ε to be low, we should maximize the proportion of the covered points and minimize the total area of the fitted ellipses.

We begin by fitting a single ellipse ($k=1$) to a connected region. If the error of the fit (ε) is above a threshold δ, we increment the number of ellipses (k) and use k-means clustering algorithm to cluster the points into k clusters. Then, an ellipse is fit to each cluster of points and the error ε is recomputed. This procedure is repeated till the value of ε falls below δ.

The steps are shown in Algorithm 2. It should be noted that, as the number of ellipses, k, is increased, A_{ell} will decrease and N_{cov} will increase.

Here, $\mathcal{C}^i$ denotes the i^{th} cluster, *FitEllipse*($\mathcal{C}$) fits an ellipse to the points belonging to cluster $\mathcal{C}$, *AreaCovered*($\mathcal{C}$,$\mathcal{E}$) computes the number of points in cluster $\mathcal{C}$ that lie inside or on the ellipse $\mathcal{E}$ and *Area*($\mathcal{E}$) computes the area of the ellipse $\mathcal{E}$.

Figure 4 shows the configurations of ellipses obtained for various values of k on a S-shaped region. As can be seen, with increasing k, the fitted ellipses



(a) k=1, $\varepsilon = 0.35$

(b) k=2, $\varepsilon = 0.28$

(c) k=4, $\varepsilon = 0.11$

Figure 4: (a)-(c) Visual representation of Algorithm 2, where k is the number of ellipses to be fit and ε is the error of the fit.

Algorithm 2: Set Cover using k-means.

```
procedure COVER($\mathcal{P}$)   ▷ $\mathcal{P}$ is the given point set
    $\delta = 0.1$   ▷ threshold
    $k = 0$   ▷ k:Number of clusters
    $\varepsilon = 1$   ▷ $\varepsilon$:Fitting error
    $N = |\mathcal{P}|$   ▷ N: no. of points in $\mathcal{P}$
    while $\varepsilon >= \delta$ do
        $k+ = 1$   ▷ Increment number of clusters
        $\mathcal{C}$ = kmeans($\mathcal{P}, k$);
        $N_{cov} = 0$   ▷ $N_{cov}$:#pixels covered
        $A_{ell} = 0$   ▷ $A_{ell}$:area of fitted ellipses
        for $i = 1 \to k$ do
            $\mathcal{E}^i$ = FitEllipse($\mathcal{C}^i$)
            $N_{cov} = N_{cov}$ + AreaCovered($\mathcal{C}^i, \mathcal{E}^i$)
            $A_{ell} = A_{ell}$ + Area($\mathcal{E}^i$)
        end for
        $\varepsilon = \frac{A_{ell}}{N_{cov}}$
    end while
end procedure
```

represent the region more accurately, which progressively decreases the error of the fit (ε).

3.4 Comparing Descriptors

Given two descriptors, the similarity score is computed by accumulating the extent of overlap of the ellipses of the corresponding bins. The overlap between two ellipses is computed using the method discussed in Section 3.1.1.

Algorithm 3 describes the steps involved in comparing two descriptors. Here $\mathcal{D}_{i,j}^{(\cdot)}$ refers to the ellipse fitted to the j^{th} connected region belonging to the i^{th} bin; *GetOverlap*($\mathcal{E}_1$,$\mathcal{E}_2$) computes the amount of overlap between the ellipses $\mathcal{E}_1$ and $\mathcal{E}_2$.

Before computing the similarity of a pair of descriptors, the ellipses comprising a descriptor are rotated along the characteristic orientation of the IR of the descriptor, which is obtained by finding the dominant orientations in the Histogram of Gradients constructed for the IR (Lowe, 2004). To achieve scale invariance, the ellipses are mapped to a circular region of unit radius.

Algorithm 3: Comparing Descriptors.

```
procedure $ov$ = COMPARE($\mathcal{D}^{(1)}, \mathcal{D}^{(2)}$)
    $n \leftarrow$ numOfBins($\mathcal{D}^{(1)}$)
    $ov \leftarrow 0$   ▷ overlap
    for $bin$=1 ... n do
        for $reg_1$=1 ... $|K_{bin}^{(1)}|$ do
            $\mathcal{E}_1 \leftarrow \mathcal{D}_{bin,reg_1}^{(1)}$
            for $reg_2$=1 ... $|K_{bin}^{(2)}|$ do
                $\mathcal{E}_2 \leftarrow \mathcal{D}_{bin,reg_2}^{(2)}$
                $ov \leftarrow ov + GetOverlap(\mathcal{E}_1, \mathcal{E}_2)$
            end for
        end for
    end for
end procedure
```

3.5 Splitting the Intensity Range

The binning of pixel intensities can be done using three methods, viz., hard binning, rank based binning and soft binning.

Hard binning: If n represents the number of bins used for quantization and the intensity range of the interest region is from a to b, then the size of each bin is $\frac{b-a}{n}$.

Rank method: Let N be the number of pixels in the connected region. The intensities of the neighborhood pixels are first sorted and the intensity of every $(N/n)^{th}$ element in the sorted list defines the bin boundary.

Soft binning: In the aforementioned binning methods, the intensity of a pixel takes an integral value and can belong to only one bin. We relax this constraint and allow the pixel to belong to more than one bin. The membership of the pixel to a particular bin is weighed by the distance from the center of the bin. It should be noted that a pixel can belong to at most two bins. The weight W for a pixel p is calculated as:

$$W(p) = 1 - \frac{|I_p - I_b|}{S_b} \tag{4}$$

Here I_p is the intensity of the pixel p, I_b denotes the center of bin and S_b denotes the bin width. The weight ranges from 0 to 1, with 0 specifying that the pixel does not belong to that bin and 1 indicating that the pixel intensity lies exactly at the bin center in which case it belongs to only that bin.

For example, say, the intensity range of a region is 101-500 and we need a 4-way split. The centers of the bins will be 150, 250, 350 and 450. A pixel with intensity 200, bordering on the boundary between bins 1 and 2, will belong to both the bins 1 and 2, with the weights being 0.5 and 0.5 respectively. A pixel with value 350 will belong to bin 3 with the weight of 1. A pixel with value 175 will belong to bins 1 and 2 and the weights will be 0.75 and 0.25 respectively. This is illustrated in Figure 5.

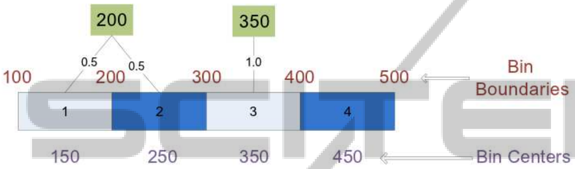


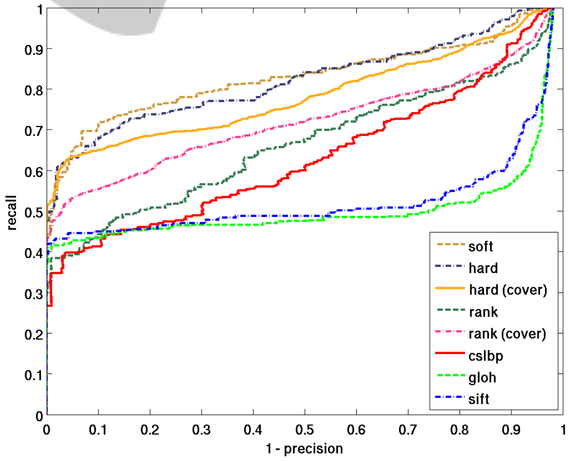
Figure 5: Illustration of the weight assignment to pixels using soft binning.

4 RESULTS

The descriptors are evaluated on Oxford data set which is used to evaluate the performance of descriptors when a scene undergoes photometric and geometric deformations. The results are presented with *recall* versus *1-precision*, as used in (Mikolajczyk and Schmid, 2005). The protocol for evaluating the descriptors is as follows. Let $\mathcal{D}^{(i)}$ denote the descriptors belonging to image *i*.

1. Given an image pair, we extract the IRs using Hessian-Affine detector from both images and compute the descriptor for each IR.
2. *Potential matches:* For a descriptor d_i in $\mathcal{D}^{(1)}$, the descriptor $d_j \in \mathcal{D}^{(2)}$ is a potential match if their similarity is greater than a threshold σ. The similarity measure is the standard Euclidean distance in case of SIFT, GLOH and CS-LBP.
3. *Ground Truth:* A descriptor $d_2 \in \mathcal{D}^{(2)}$ is said to correspond to $d_1 \in \mathcal{D}^{(1)}$ if the overlap of IR of d_1 with that of d_2 more than 50% (as outlined in Section 3.1.1).
4. *Correct matches:* The overlap of the IR corresponding to $d_i \in \mathcal{D}^{(1)}$ with those of the potential matches in $\mathcal{D}^{(2)}$ is computed, using the method outlined in Section 3.1.1 and the match is said to be correct if the overlap is greater than 50%. It should be noted that the correct matches for a descriptor $d_i \in \mathcal{D}^{(1)}$ are those descriptors $d_j \in \mathcal{D}^{(2)}$ that are present in both the set of correspondences and the set of potential matches.
5. Now the *recall* is given as the ratio of the number of correct matches to the total number of correspondences and *precision* is given as the proportion of correct matches among the potential matches. For each value of σ, we compute *(1-precision)* and *recall*, which are then used to generate the curves.

In our experiments, we compare the performance of our descriptor with that of SIFT, GLOH[2] and CS-LBP. The number of bins is fixed at 8 while constructing the descriptors using all the three binning methods. The region covering algorithm (*Cover*) outlined in Algorithm 2, was used with rank and hard binning methods; the threshold δ for the ellipse fitting was set at 0.1 and the maximum number of iterations for k-means was set at 20. Figure 6 shows the performance of the descriptors for various image deformations. It should be noted that the region covering method was used with rank and hard binning methods since the connected regions belonging to the different bins can be clearly demarcated with these binning methods. The time taken for constructing the descriptor ranged from 4.1s for 2 bins to 4.5s for 8 bins, where SIFT and GLOH took 1.2s.



(a) Scaling+Rotation

Figure 6: Performance of the descriptors in the presence of geometric deformation. Number of bins for our method is fixed at 8 while constructing descriptors using all three binning methods.

Figure 6 shows the performance of the descriptors in case of geometric deformations, i.e., scaling and rotation changes. The hard binning method performs

[2] Binaries for SIFT and GLOH were obtained from http://www.robots.ox.ac.uk/~vgg/research/affine/descriptors.html #binaries

better than the rank based binning, since the intensities of the IRs are not distorted under these deformations. In this case, all the three binning methods perform better than the existing state of the art methods.

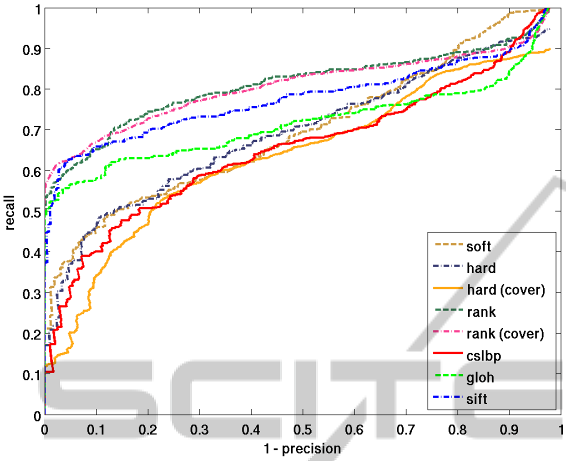


Figure 7: Performance of the descriptors in the presence of blur. Number of bins for our method is fixed at 8 while constructing descriptors using all three binning methods.

Figure 7 shows the performance in case of blur. The intensity maps of the corresponding IRs are considerably more distorted when compared to geometric deformations. As is evident from the figure, rank based binning works better than the other binning methods. This can be attributed to the fact that the order of the pixel intensities is not affected to a great degree under these deformations and thus the bin boundaries and the bin labels are less distorted when compared to the other binning methods.

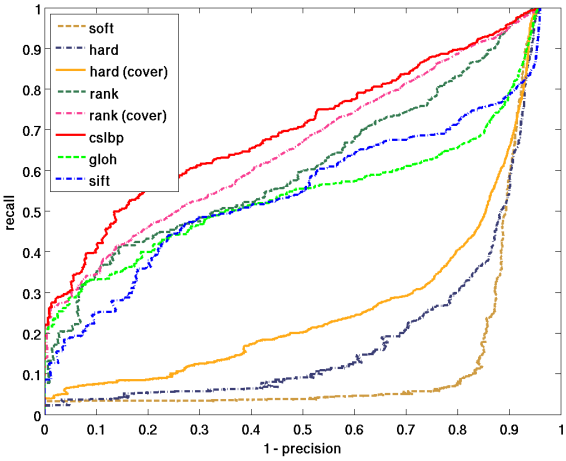


Figure 8: Performance of the descriptors in the presence of nonlinear illumination changes. Number of bins for our method is fixed at 8 while constructing descriptors using all three binning methods.

Figure 8 shows the effect of nonlinear illumination changes on the performance of the descriptors. In the presence of illumination changes, CS-LBP is more robust than all of the descriptors, but then in this case, the performance of our descriptor is at par.

In conclusion, the rank based binning method with region covering performs better than its naive counterpart and the other binning methods. It was found that the accuracy of the proposed descriptor increases as the number of bins (n) is increased from 2 bins and saturates when the value is 8. This is because the descriptor is a coarse representation of the underlying intensity map when n is 2 and the representation becomes finer with increasing n. But increasing n beyond 8 was detrimental to the performance, because the width of the individual bins decreases with increasing n and a slight distortion of the intensities changes the bin labels of the pixels and deforms the n map.

It is interesting to note that, in Figure 2, the proportion of the pixels retaining the bin label is high when the n is low. But, such n-maps are over generalized representations of the underlying IR and impair the discriminative ability of the descriptor.

5 SUMMARY AND CONCLUSIONS

In this paper, we have proposed a novel method for constructing image descriptors, using intensity binning, which involves quantization of the intensity map of the interest regions and fitting ellipses to each connected region of the bins obtained. We also proposed a better representation of such binned intensity maps using k-means. These approaches were evaluated on images with commonly occurring image deformations. The experiments show that the proposed descriptor is robust to photometric and geometric deformations and outperforms the current state of the art methods. As we had shown, the optimal number of bins for quantization was chosen empirically. A more principled way would be to choose the number of bins by closer inspection of topography of the regions obtained across multiple quantization levels, which would be a promising direction for improving the proposed method.

REFERENCES

Agarwal, S., Snavely, N., Simon, I., Seitz, S., and Szeliski, R. (2009). Building rome in a day. In *IEEE International Conference on Computer Vision*, pages 72–79.

Baumberg, A. (2000). Reliable feature matching across widely separated views. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 774–781.

Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding*, 110:346–359.

Berg, A., Berg, T., and Malik, J. (2005). Shape matching and object recognition using low distortion correspondences. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 26–33.

Berg, A. and Malik, J. (2001). Geometric blur for template matching. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–607 – I–614 vol.1.

Brown, M., Hua, G., and Winder, S. (2011). Discriminative learning of local image descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):43–57.

Brown, M. and Lowe, D. G. (2003). Recognising panoramas. In *IEEE International Conference on Computer Vision*, pages 1218–1225.

Chen, J., Shan, S., He, C., Zhao, G., Pietikainen, M., Chen, X., and Gao, W. (2010). WLD: A robust local image descriptor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1705 –1720.

Cheng, H., Liu, Z., Zheng, N., and Yang, J. (2008). A deformable local image descriptor. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1 –8.

Fergus, R., Perona, P., and Zisserman, A. (2003). Object class recognition by unsupervised scale-invariant learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 264–271.

Freeman, W. and Adelson, E. (1991). The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):891–906.

Harris, C. and Stephens, M. (1988). A combined corner and edge detector. In *Alvey vision conference*, pages 147–152.

Heikkil, M., Pietikinen, M., and Schmid, C. (2009). Description of interest regions with local binary patterns. *Pattern Recognition*, 42(3):425–436.

Johnson, A. E. and Hebert, M. (1997). Recognizing objects by matching oriented points. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 684–689.

Ke, Y. and Sukthankar, R. (2004). PCA-SIFT: A more distinctive representation for local image descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Kennedy, L. S. and Naaman, M. (2008). Generating diverse and representative image search results for landmarks. In *International Conference on World Wide Web*, pages 297–306.

Lazebnik, S., Schmid, C., and Ponce, J. (2005). A sparse texture representation using local affine regions. *PAMI*, 27(8):1265–1278.

Leibe, B., Leonardis, A., and Schiele, B. (2008). Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision*, 77(1-3):259–289.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110.

Mikolajczyk, K. and Schmid, C. (2001). Indexing based on scale invariant interest points. In *IEEE International Conference on Computer Vision*, pages 525–531.

Mikolajczyk, K. and Schmid, C. (2005). A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630.

Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., and Gool, L. (2005). A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1):43–72.

Nister, D. and Stewenius, H. (2006). Scalable recognition with a vocabulary tree. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2161–2168.

Snavely, N., Seitz, S., and Szeliski, R. (2008). Modeling the world from internet photo collections. *International Journal of Computer Vision*, 80(2):189–210.

Tuytelaars, T. and Mikolajczyk, K. (2008). Local invariant feature detectors: A survey. *Foundations and Trends in Computer Graphics and Vision*, 3(3):177–280.

Tuytelaars, T. and Van Gool, L. (2004). Matching widely separated views based on affine invariant regions. *International Journal of Computer Vision*, 59(1):61–85.

Viola, P. and Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision*, 57:137–154.

Zhang, J., Marszalek, M., Lazebnik, S., and Schmid, C. (2007). Local features and kernels for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision*, 73:213–238.