An Intensity-augmented Ordinal Measure for Visual Correspondence

Anurag Mittal
Real Time Vision and Modeling*
Siemens Corporate Research
Princeton, NJ 08540

Visvanathan Ramesh
Real Time Vision and Modeling
Siemens Corporate Research
Princeton, NJ 08540
visvanathan.ramesh@siemens.com

Abstract

Determining the correspondence of image patches is one of the most important problems in Computer Vision. When the intensity space is variant due to several factors such as the camera gain or gamma correction, one needs methods that are robust to such transformations. While the most common assumption is that of a linear transformation, a more general assumption is that the change is monotonic. Therefore, methods have been developed previously that work on the rankings between different pixels as opposed to the intensities themselves. In this paper, we develop a new matching method that improves upon existing methods by using a combination of intensity and rank information. The method considers the difference in the intensities of the changed pixels in order to achieve greater robustness to Gaussian noise. Furthermore, only uncorrelated order changes are considered, which makes the method robust to changes in a single or a few pixels. These properties make the algorithm quite robust to different types of noise and other artifacts such as camera shake or image compression. Experiments illustrate the potential of the approach in several different applications such as change detection and feature matching.

1. Introduction

Determining the correspondence of image patches is one of the most important problems in Computer Vision, with applications to stereo matching, change detection, optical flow, image registration etc. In many of these applications, such matching has to be performed under many possible intensity changes occuring due to the change in camera gain and offset, gamma correction, illumination changes etc.

In order to achieve invariance to such factors, most methods assume a linear transformation model such that the same change in the illumination in different nearby pixels or color components creates the same proportion of change in the intensity observed at these pixels. Normalized cross-correlation is one such commonly considered distance measure between two image patches that is invariant to a linear change in intensity. More complex methods utilize a variety of different filters and techniques in order to achieve invariance to a linear intensity change[15, 12, 2, 5, 7, 8, 16, 18, 19, 17]. The normalization can also be performed in the spectral space by methods such as normalized color, including its variations that utilize a robust matching technique[10].

However, it is well-known[1, 21, 6] that many image transformations are non-linear in nature: gamma correction causes non-linearity, the camera response function is not linear near saturation and low-light, small specular reflection or dust/rain/snow peckles can change some pixels, and different parts of an object may be illuminated differently. One method that has been considered to handle such changes in the visual space is that of mutual information[20] that can handle a complete change in the image intensities. While such an approach can be used[14], the most restrictive assumption that is able to handle the actual image transformation should be used for best performance. An assumption that is more appropriate in many circumstances is that the changes are monotonic. Many methods have utilized this assumption in order to achieve techniques that are more robust under these more general changes. Most of these methods transform the feature space such that only the "order" of a particular pixel in relation to its neighbors is considered. The census transform[22] 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[4] improved upon such measure by a carefully designed distance between two rank permutations. While a single pixel error can cause disproportionate error in the census algorithm, this method counts such changes

*The author is currently with the Department of Computer Science and Engg., Indian Institute of Technology Madras, Chennai, INDIA - 600036. He may be reached at amittal@cse.iitm.ernet.in.

0-7695-2646-2/06 $20.00 (c) 2006 IEEE

only once. Thus, their method is more robust to random pixel changes due to specularities, camera shake and salt and pepper noise.

While such approaches are robust to monotonic changes in the intensity and are relatively stable against random changes in the signal, the presence of Gaussian noise can severely deteriorate the performance of such methods. The reason is that, under Gaussian noise, there is a high probability of an order flip when the intensities of the two pixels are close to each other.

In this paper, we develop a new image matching method that improves upon such methods by using a combination of intensity and rank information in order to achieve greater robustness to Gaussian noise while maintaining invariance to monotonic intensity changes. The method looks at rank changes amongst n pixels in a patch. A change of order between pixels that are far from each other (in intensity) is given a greater 'penalty' compared to an order change between "closer" pixels. This makes the method relatively stable to Gaussian noise. Furthermore, while C_2^n orders (and possible order changes) exist between n pixel variables, we develop a novel technique to choose only the most representative order changes without "double counting" the changes occuring due to an error in a single pixel. This property enables the algorithm to be robust to salt and pepper noise, camera shake and specular reflections. Experiments illustrate the improvement achieved by the algorithm compared to prior approaches.

2. Designing a Robust Change Measure

As noted above, invariance to a monotonic intensity transformation is a desirable property of a robust matching method. Monotonicity in the observations implies that the order of such observations is still maintained. Thus, one can use this property in order to develop a measure of change that detects changes in such order information.

Methods exist in the literature that utilize such order information in order to determine changes[4, 3, 22, 11]. These methods first transform the original patch into a rank space that only considers the rankings between pixels. Measures are then developed to determine changes between two patches based on a change in the rankings.

When the two patches have n pixels each, there are C_2^n different pairs, among which one can compute an order change. Traditional methods such as Spearman's ρ and Kendall's τ [13, 9]have looked at measures that compute an order change among all such pairs, and determine a score for a rank change based on some kind of average, or proportion of the rankings that undergo a change. Such measures, however, severely penalize change in even a single observation and count all the possibly n order changes due to a single pixel even though such changes are correlated. such measures are not too robust, and In order to address this drawback, Bhat and Nayar[4] proposed a method that is relatively invariant to such changes. They devise a clever method to count, for each pixel, the number of elements less than it that are out of position and determine the maximum of such count over all pixels. This method counts one pixel change only once and works fairly well in practice.

However, while such methods are very robust to monotonic changes and are designed to be pretty robust to random pixel changes (the so-called salt and pepper noise occuring due to the camera, dust particles, rain, snow, specular reflections etc.), they are not very robust to Gaussian noise since even a small amount of Gaussian noise can completely change the rankings between pixels that are not far from each other in intensity. Such drawback occurs since they do not consider the actual intensities at all during the matching process. Therefore, one property of a desired algorithm is that the change measure should be a function of the difference between the pixels that undergo an order change, measuring a higher change as such difference increases. In this paper, we develop a strategy that yields a change measure that is robust to such small changes in the input data, while also maintaining most of the advantages of such order-based methods.

3. Our Change Measure

Let I_1 and I_2 be two windows that are to be compared. Then, let π_1^i be the rank of I_1^i among the I_1 data, and π_2^i be the rank of I_2^i among the I_2 data. Let us consider a class of sets S^c where any set $S \in S^c$ consists of a set of (i, j) indices, where i and j represent the i-th and j-th elements of windows I_1 and I_2:

$$S \subset \{(i, j) : i, j \in 1 \dots n\} \tag{1}$$

Now, define a *flip* set $S_{FL} \in S^c$ such that:

$$S_{FL} = \{(i, j) : Sgn(\pi_1^i - \pi_1^j) \neq Sgn(\pi_2^i - \pi_2^j)\} \tag{2}$$

where Sgn signifies the sign function. In simple words, S_{FL} is the set of index pairs (i, j) that have their orders flipped from one window to the other. The basic idea behind defining this set is that it specifies all the order changes that have occured between the two sets and can thus be used to develop a change measure. For instance, Kendall's τ simply counts the number of elements in S_{FL}. Similarly, the census algorithm[22] and the algorithm by Bhat and Nayar[4] derive useful measures by working on this set directly without regard to the intensity values. In our algorithm, we go beyond the rank of the elements and develop a distance measure using the underlying intensities.

In order to do so, let us further define a distance function f on the difference between two intensity values. Then, a *forward* distance function on sets $S \in S^c$ may be defined thus:

$$D_f^1(S) = \Sigma_{(i,j)\in S} f(I_1^i - I_1^j) \tag{3}$$

A *backward* distance function $D_f^2(S)$ is defined similarly. When applied to sets containing *flipped* pairs of elements, this distance measures not the number of pairs that are flipped, but the sum of the intensity differences between the flipped elements. Thus, a flip in elements that are close to each other is counted less compared to a flip in elements that are far from each other. This property helps in making the distance measure robust to Gaussian noise that can easily cause a flip in elements that are close to each other in intensity. However, if this distance measure is used directly on S_{FL}, then the distance measure would be very sensitive to change in even a single pixel as all pairs containing this pixel would be counted. All such *flip* pairs are correlated with each other and hence we design our detection measure so as to avoid such double counting.

In order to achieve this property, let us define a property which we call *elemental uniqueness*. Any set S with this property has any element occuring only once (i.e. there do not exist $(i_1, j_1), (i_2, j_2) \in S$ such that $i_1 = i_2$ or $j_1 = j_2$). Then, we define our *forward* distance measure as:

$$d_f^1 = \max_{\substack{S \subset S_{FL}, and \\ \text{S is } \textit{elementally unique}}} D_f^1(S) \tag{4}$$

Thus, we design our measure in such a way that it maximizes the sum of elemental differences (according to the function f) among uncorrelated flipped pairs.

We then normalize the distance so obtained using the maximum possible value of such measure since the magnitude of the distance can vary significantly depending on the amount of texture in the patch. Using the *forward* distance measure, a *forward* detection measure γ_f^1 may thus be defined:

$$\gamma_f^1 = \frac{d_f^1}{dmax_f^1} \tag{5}$$

where $dmax_f^1$ is the maximum possible measure of the distance and may be computed as:

$$dmax_f^1 = \sum_{i=1}^{n/2} f(I_1^i - I_1^{n-i}) \tag{6}$$

The *backward* detection measure γ_f^2 is defined similarly:

$$\gamma_f^2 = \frac{d_f^2}{dmax_f^2} \tag{7}$$

The distances (and detection measures) from the two directions are different since the intensities from the two sides are different and even the *maximal* pairings that are selected are different from the two sides. Since the orders must be preserved from both sides, one could take the maximum of the two detection measures $max(\gamma_f^1, \gamma_f^2)$ to determine the change between two patches.

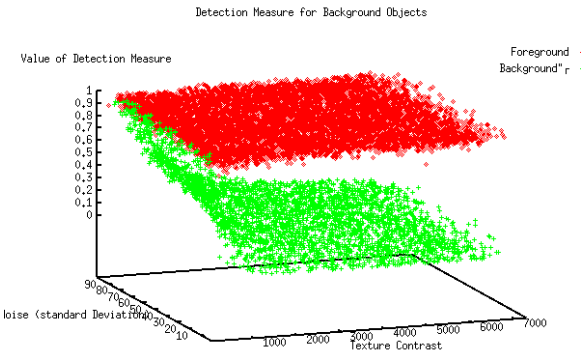


Figure 1. The response of the detection measure as a function of the texture and noise levels. The experiments were conducted on random image patches and the background was perturbed by a random multiplicative factor. Note that the discriminability of the method goes down as a function of the texture content (dmax) of the patch.

Another consideration, however, is that the ranking of the side with the higher texture contrast (as measured by $dmax_f$) is more stable (the response of the detection measure to foreground and background as a function of the texture and noise is shown in Fig. 1). Hence, we found that it is more stable to simply select the change measure from the side with the higher texture contrast value. Thus, we define our detection measure γ_f as:

$$\gamma_f = \gamma_f^{argmax(dmax_f^1, dmax_f^2)} \tag{8}$$

Furthermore, if both $dmax_f^1$ and $dmax_f^2$ are below a certain threshold, we say that the comparison is ill-defined.

4. Efficient Computation of the Detection Measure

The detection measure defined above is extremely expensive to compute in the naive fashion. For a certain class of functions f which are interesting to us, however, this measure can be computed very efficiently. Let us assume that $f(x) = |x|$. Then, the following algorithm computes the distance d_f^1 in order $O(n^2)$ time, where n is the number of pixels in the patch.

First, consider the element e_1 with rank 1 in I_1 (assuming that I_1 is pre-sorted in ascending order). Now, we consider all elements that have a rank change with this element in the other set. Then, we find the highest ranked element e_{h1} that has an order flip with this pixel. Now, it can be shown that for both of these elements, this order flip is the flip with the highest inter-element difference. To see this, note that for element e_1, we have found the highest - flipping counterpart by construction. For the paired element e_{h1} that was found, e_1 is obviously the element lower than e_{h1} in rank that has the highest elemental difference. The

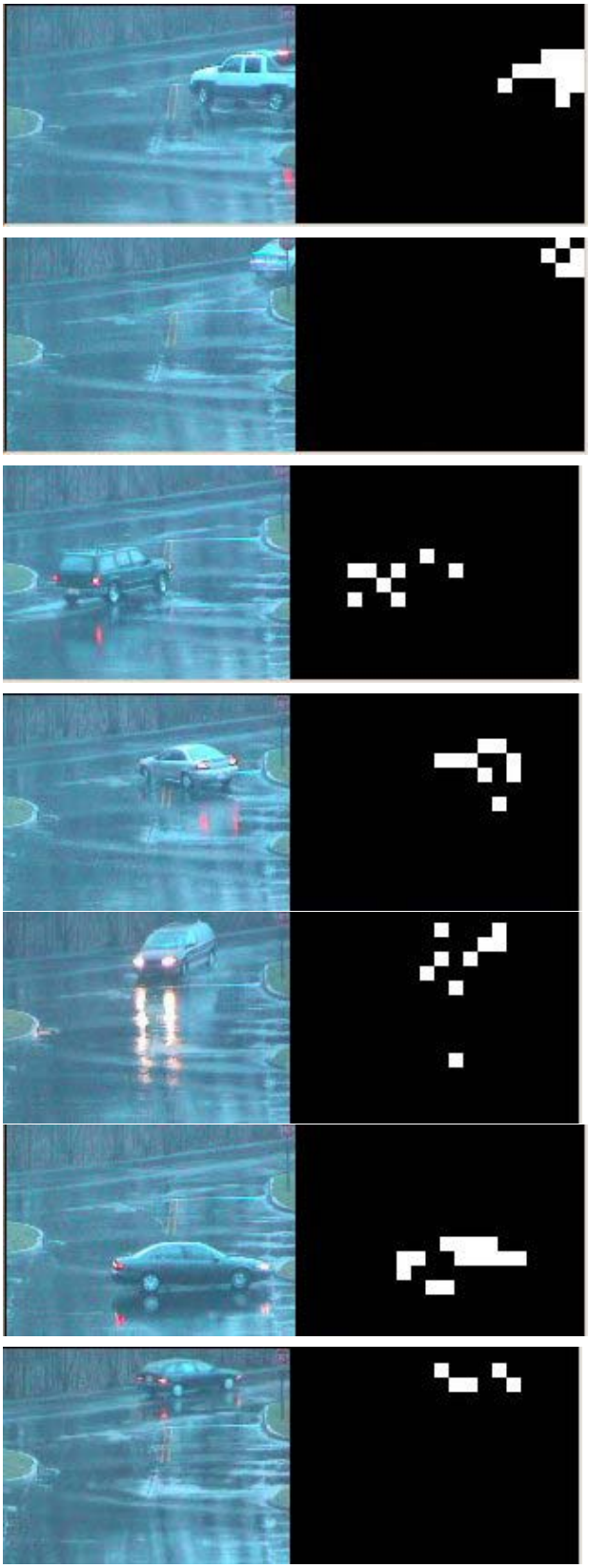


Figure 2. Some detection results from our algorithm on a rain sequence. Note that some regions are not detected as both the background and foreground are homogenous.

Figure 3. Detection Results from Bhat and Nayar[4] on the rain sequence. Each image in this figure may be compared with the corresponding figure in Fig. 2.

only elements left to consider are the ones greater than it in rank. However, if there was any element e_{h2} greater than this element with which it had a flip, then that element e_{h2} would have a flip with e_1 element as well (since then, $rank_{I_2}(e_{h2}) < rank_{I_2}(e_{h1}) < rank_{I_2}(e_1)$). If this was the case, however, we would have found e_{h2} during our search for the highest element and not e_{h1}, thus contradicting the definition of e_{h1}. Hence, for both e_1 and e_{h1}, this is the highest difference pairing.

In the next step, we remove both these elements from consideration so as not to repeat them (*the elemental uniqueness* constraint). Then, the above step for the rest of the element list is repeated. The sum of the elemental differences for all such pairs thus found is the detection measure d_f^1. To prove that the algorithm finds the pairings with the highest sum of elemental differences, it is sufficient to show that the elemental pairing (e_1, e_{h1}) must be present in the *maximal* set. The rest of the proof follows by induction. To show this, suppose this pairing is not present in the set and instead, these elements are paired with some other elements, i.e. the pairs (e_1, e_i) and (e_j, e_{h1}) are present. However, since $f(x) = |x|$, it can directly be seen that $f(I(e_{h1}) - I(e_1)) > f(I(e_i) - I(e_1)) + f(I(e_{h1}) - I(e_j))$ if $I(e_j) > I(e_i)$, and $f(I(e_{h1}) - I(e_1)) + f(I(e_i) - I(e_j)) = f(I(e_i) - I(e_1)) + f(I(e_{h1}) - I(e_j))$ if $I(e_i) > I(e_j)$. In the first case, both the pairs (e_1, e_i) and (e_j, e_{h1}) could be replaced with (e_1, e_{h1}) with a higher elemental difference value. In the latter case, one could as well select (e_1, e_{h1}) and (e_j, e_i) rather than (e_1, e_i) and (e_j, e_{h1}) with the same elemental difference. Thus, selecting (e_1, e_{h1}) as opposed to (e_1, e_i) and (e_j, e_{h1}) can only increase the difference value and not decrease it. Thus, by induction, one can show that the algorithm produces the sum of the maximum elemental differences d_f^1. The result is also true for all functions f that have a positive second derivative everywhere (for e.g. such $f(x) = x^2$). The proof for this more general result is very similar to the one above.

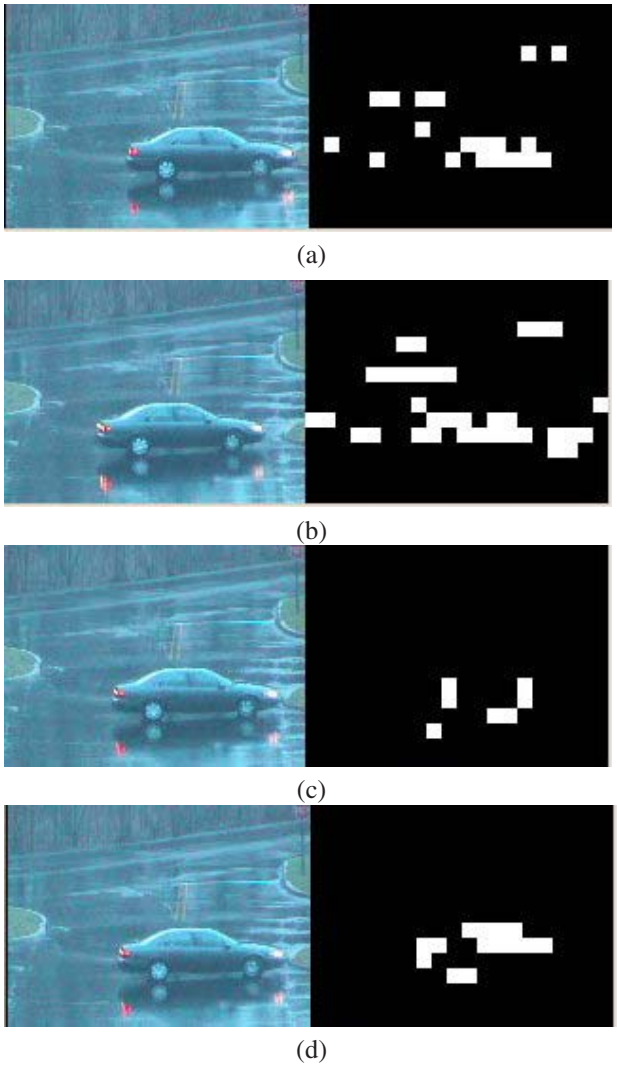


Figure 4. Comparative results for four algorithms. (a): Normalized Cross-correlation, (b) Census Algorithm, (c) Bhat-Nayar, and (d) Our approach.

5. Applications and Experiments

Our basic matching technique can be applied to a variety of different applications. Here, we will illustrate our algorithm with respect to two applications: change detection and feature matching.

5.1. Change Detection

First, we applied our matching technique to the common application of background subtraction. An image without any objects was used as the background and all subsequent frames were compared with this image in order to detect foreground objects. First, we illustrate our results on an outdoor sequence where one wants to detect moving cars in the presence of heavy rain. The scene contains a lot of noise due to the falling rain, reflection of the cars on the ground and image blur due to the weather conditions. At the same time, the sequence was captured via an IP camera that transmits JPEG images. This also introduces some compression noise. Our algorithm worked very well on this sequence and sample results are presented in Fig. 2. The only places where the algorithm failed to detect were regions where the foreground as well as the background were homogenous. Failure to detect such regions is an inherent drawback of all methods that are illumination invariant. In order to improve the detection in such regions (though breaking the illumination-invariance), it is possible to compare the means of the two patches and then detect an object if the difference of the means is above a certain threshold.

Figure 5. Results of our algorithm on a sequence containing people on a subway platform. Note that the light was switched on in the right top corner of the video.

We did not use this criteria for the results in this paper.

The same sequence was then processed via three other algorithms: normalized cross-correlation, the Census algorithm [22] and the algorithm by Bhat and Nayar [4]. We found that normalized cross-correlation and the census algorithms could not detect anything at all when the thresholds were set such that the false alarm rate was equal to what we obtained for the used settings of our algorithm. Only when the false alarm rates were very high could the two algorithms detect some objects. Some sample comparative results are presented in Fig. 4. The algorithm by Bhat and Nayar did detect some objects with the same false alarm rate, but the detection rate was lower than our algorithm. The detailed results from this algorithm on this sequence are presented in Fig. 3. This may be compared image-by-image to Fig. 2. The presence of rain introduces some salt and pepper noise and hence, it was expected that the normalized cross-correlation would not work very well. At the same time, there is some Gaussian noise and blur, which would throw off the census algorithm. Bhat-Nayar appears to be more robust to these types of noise.

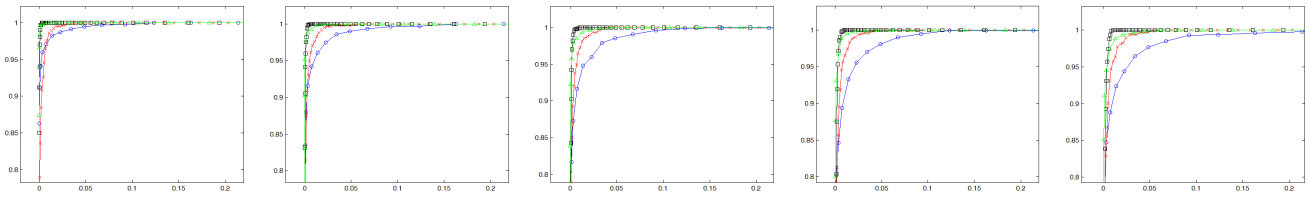
Next, we show the results of our algorithm on a sequence from a subway station, where the algorithm was used to detect any foreground objects. The scene contains many illumination changes due to shadows, light from the train and camera gain changes (which can be very severe when some object becomes very big in the scene). At the same time, there were specularities in the scene due to the reflection of the train lights from floor and there was significant noise in the video due to MPEG compression effects. The method was found to be very robust to all of such illumination changes and had a high detection rate. However, as with the previous scene, the homogenous regions are often missed.

5.2. Feature Matching

In order to evaluate our algorithm with respect to feature matching, we took a set of test images. Then, random patches of size 10×10 were selected from these images and tested against some other random patches in order to determine the false match (detection) rate. The same patch was then compared against itself after adding some particular types of perturbations. This gives us the detection rate. Since textures with low contrast cannot be compared against each other satisfactorily, the comparisons were only considered valid if both the compared patches had a significant contrast. For these same valid point matches, the algorithm was tested against three other matching methods: normal cross correlation, the census algorithm[22] and Bhat-Nayar[4]. The two latter algorithms are most comparable to ours because of their invariance to a monotonic change and normalized cross-correlation was chosen because of its popularity and as a representative of methods that assume a linear intensity transformation model.

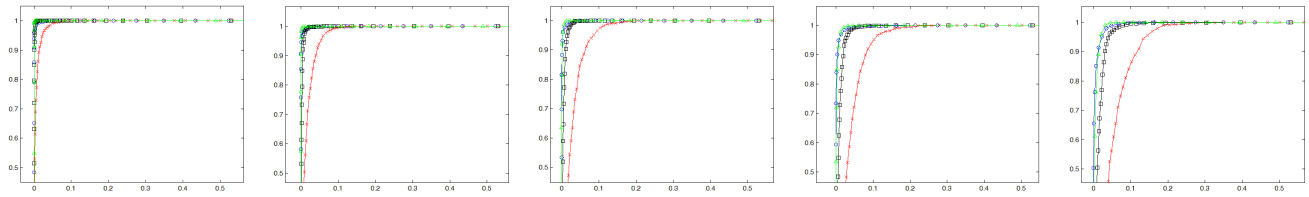
First, we applied two types of noise to the data and computed the *ROC* curves for all the algorithms. Fig. 6 shows the *ROC* curves as a function of the amount of Gaussian noise added. Our method performed better than all the other methods tested in this experiment. As stated earlier, ranking-based methods cannot deal satisfactorily with Gaussian noise when some pixels have intensities that are close to each other and can cause a rank change with only a small amount of noise. Thus, this under-performance was expected. The underperformance of normalized cross-correlation is perhaps due to the over-generalization of a linear model.

On the other hand, when salt and pepper noise was added (Fig. 7), the rank-based methods out-performed the others. This was expected since the rank-based methods simply count the order changes and hence have a better tolerance to salt and pepper noise. Normalized cross-correlation performed the worst, while our method was in-between the two



(a) $\sigma = 10$ (b) $\sigma = 20$ (c) $\sigma = 30$ (d) $\sigma = 40$ (e) $\sigma = 50$

Figure 6. The *ROC* curves for different Gaussian noise levels. In all figures, Black with squares represents our method, Red with crosses represents Normalized Cross Correlation, Blue with circles represents Bhat-Nayar and Green with triangles represents the Census algorithm. x-axis is the false detection rate and y-axis is the correct detection rate.



(a) change = 5 % (b) change = 10 % (c) change = 15 % (d) change = 20 % (e) change = 25 %

Figure 7. The *ROC* curves for different percentages of random pixel changes (salt and pepper noise). The legends are the same as in the previous figure.

types of methods.

In practice, both these types of noise are present and hence we tested the algorithms in the presence of both types of noise. A typical scenario was used and as expected, it was found that our algorithm was able to outperform the others in this scenario (Fig. 8 (a)).

We also tested the performance of the algorithms when the images were transformed via a gain change (Fig. 8). This is a typical scenario as the gain can change dramatically due to the change in lighting, or presence of a large object in front of the camera. In general, our algorithm worked much better than others under a change in gain. One of the main reasons could be that a gain change can typically cause non-monotonicity, especially near the two ends of the intensity scale, throwing off linear methods such as normalized cross-correlation. At the same time, Gaussian noise can significantly affect the existing rank-based methods at the two ends of the camera range. Some sample images under such gain change are shown in Fig.s 8 (d), (e) and (f). Note the loss of information due to saturation and low light in several regions. Finally, the algorithms were tested in the presence of the common problems of camera shake and image blur. In the presence of both Gaussian and salt-and-pepper noise, the *ROC* curves obtained are shown in Fig. 9.

6. Conclusion

In this paper, we have presented a method for feature matching that is invariant to monotonic intensity changes. While traditional rank-based methods neglect the intensity completely, we have found that considering the intensity as well makes the matching more robust to Gaussian noise that is normally present. We demonstrated the application of the method to the problems of change detection and feature matching. Other possible applications include illumination-invariant stereo matching, optical flow computation and image registration.

References

[1] K. Barnard and B. Funt. Camera characterization for color research. *Color Research and Application*, 27(3):153–164, 2002.

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

[3] D. Bhat and S. Nayar. Ordinal measures for visual correspondence. In *CVPR*, pages 351–357, 1996.

[4] D. Bhat and S. Nayar. Ordinal measures for image correspondence. *PAMI*, 20(4):415–423, Apr. 1998.

[5] G. Carneiro and A. Jepson. Multi-scale phase-based local features. In *CVPR*, pages I: 736–743, 2003.

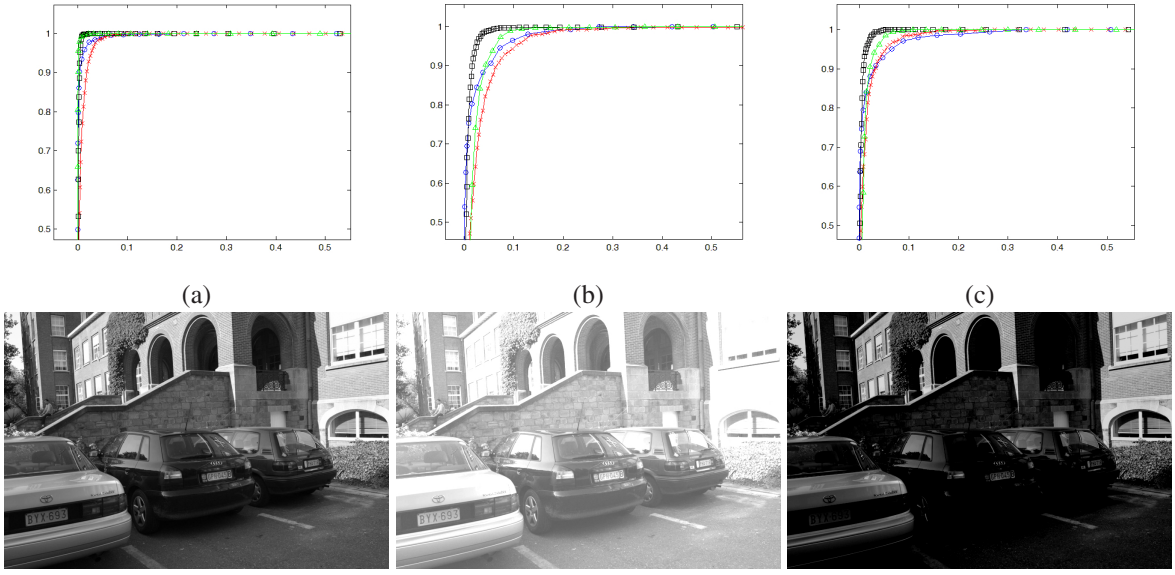
[6] D. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice-Hall, 2003.

[7] W. Freeman and E. Adelson. The design and use of steerable filters. *PAMI*, 13(9):891–906, September 1991.

[8] D. Gabor. Theory of communication. *Journal I.E.E.*, 3(93):429–457, 1946.

[9] R. Gideon and R. A. Hollister. A rank correlation co-efficient. *Journal of the Americal Statistical Association*, 82(398):656–666, 1987.

[10] M. Greiffenhagen, V. Ramesh, D. Comaniciu, and H. Niemann. Statistical modeling and performance characterization of



(a) (b) (c)

(d) (e) (f)

Figure 8. The *ROC* curves for different gain levels and a typical scenario of a Gaussian noise with $\sigma = 20$ and salt-and-pepper noise of 5 %. (a) is the curve without any gain change while (b) and (c) represent an increase and decrease in the gain respectively. Corresponding sample images with these gain settings are shown in (d), (e) and (f). The legends for the *ROC* curves are the same as in the previous figures.

a real-time dual camera surveillance system. In *CVPR*, pages II:335–342, Hilton Head, SC, 2000.

[11] M. Heikkil, M. Pietikainen, and J. Heikkila. A texture-based method for detecting moving objects. In *BMVC*, 2004.

[12] Y. Ke and R. Sukthankar. Pca-sift: a more distinctive representation for local image descriptors. In *CVPR*, pages II: 506–513, 2004.

[13] M. Kendall and J. Gibbons. *Rank Correlation Methods*. Edward Arnold, fifth edition, 1990.

[14] J. Kim, V. Kolmogorov, and R. Zabih. Visual correspondence using energy minimization and mutual information. In *ICCV*, pages 1033–1040, 2003.

[15] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, Nov. 2004.

[16] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *ECCV*, page I: 128 ff., 2002.

[17] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *PAMI*, 27(10):1615–1630, Oct. 2005.

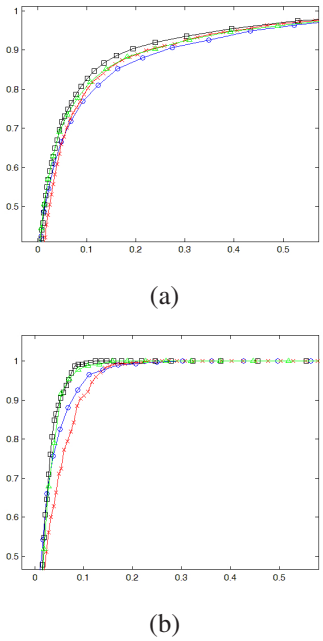
[18] T. Tuytelaars and L. Van Gool. Matching widely separated views based on affine invariant regions. *IJCV*, 59(1):61–85, Aug. 2004.

[19] M. Varma and A. Zisserman. A statistical approach to texture classification from single images. *IJCV*, 62(1-2):61–81, Apr. 2005.

[20] P. Viola and W. M. WellsIII. Alignment by maximization of mutual information,. In *ICCV*, pages 16–23, Boston, MA, 1995.

[21] P. Vora, J. Farrell, J. D. Tietz, and D. H. Brainard. Digital color cameras - 1 and 2 - response models. *Hewlett-Packard Laboratory, Technical Report*, 2001.

[22] R. Zabih and J. Woodfill. Non-parametric local transforms fo computing visual correspondence. In *ECCV*, pages 151–158, 1994.



(a)

(b)

Figure 9. The *ROC* curves for (a) a camera shake of 2 pixels (in a 10×10 block) in vertical and horizontal directions, and (b) image blur. A Gaussian noise of $\sigma = 20$ and salt-and-pepper noise of 5 % were also assumed. The legends are the same as in the previous figures.