(1) Title: Towards Improved Ranking Metrics

(2) Authors: Nicu Sebe, Michael S. Lew, Dionysius P. Huijsmans

(3) Keywords: maximum likelihood, ranking metrics, content-based retrieval, color indexing, stereo matching, motion tracking

(4) Contact author:

Nicu Sebe
Leiden Institute of Advanced Computer Science
Niels Bohrweg 1
2300 CA Leiden
The Netherlands
E-mail: nicu@liacs.nl
Fax:+31-71-527-6985
Tel:+31-71-527-7050

(5) Other authors:

Michael S. Lew
Leiden Institute of Advanced Computer Science
Niels Bohrweg 1
2300 CA Leiden
The Netherlands
E-mail: mlew@liacs.nl
Fax:+31-71-527-6985
Tel:+31-71-527-7034

Dionysius P. Huijsmans
Leiden Institute of Advanced Computer Science
Niels Bohrweg 1
2300 CA Leiden
The Netherlands
E-mail: huijsman@liacs.nl
Fax:+31-71-527-6985
Tel:+31-71-527-7052

Towards Improved Ranking Metrics

Nicu Sebe Michael S. Lew Dionysius P. Huijsmans

Leiden Institute of Advanced Computer Science

Niels Bohrweg 1, 2333 CA, Leiden, The Netherlands

{nicu mlew huijsman}@liacs.nl

Keywords: maximum likelihood, ranking metrics, content-based retrieval, color indexing, stereo matching, motion tracking

Abstract

In many computer vision algorithms, a metric or similarity measure is used to determine the distance between two features. The Euclidean or SSD (sum of the squared differences) metric is prevalent and justified from a maximum likelihood perspective when the additive noise distribution is Gaussian. Based on real noise distributions measured from international test sets, we have found that the Gaussian noise distribution assumption is often invalid. This implies that other metrics, which have distributions closer to the real noise distribution, should be used. In this paper we consider three different applications: content-based retrieval in image databases, stereo matching, and motion tracking. In each of them, we experiment with different modeling functions for the noise distribution and compute the accuracy of the methods using the corresponding distance measures. In our experiments, we compared the SSD metric, the SAD (sum of the absolute differences) metric, the Cauchy metric, and the Kullback relative information. For several algorithms from the research literature which used the SSD or SAD, we showed that greater accuracy could be obtained by using the Cauchy metric instead.

1 Introduction

At the core of many algorithms in computer vision is the metric or similarity measure used to determine the distance between two features. The SSD (sum of the squared differences) and

SAD (sum of the absolute differences) are the most commonly used metrics. This brings to mind several questions. First, under what conditions should one use the SSD versus the SAD? From a maximum likelihood perspective, it is well known that the SSD is justified when the additive noise distribution is Gaussian. The SAD is justified when the additive noise distribution is Exponential (double or two-sided exponential). Therefore, one can determine which metric to use by checking if the real noise distribution is closer to the Gaussian or the Exponential. This leads to the second question: What distance measure do we use in comparing the real noise distribution to the best fit Gaussian or Exponential distributions? This is not an easy question to answer because the choice of the distance measure will bias the comparison. In practice, the Chi-square test is frequently used and, since we have not found a better solution, we used it for comparing the distributions.

The common assumption is that the real noise distribution should fit either the Gaussian or the Exponential, but what if this assumption is invalid? What if there is another distribution which fits the real noise distribution better than the Gaussian or the Exponential? It is precisely this question which we examined in this paper. Toward answering this question, we have endeavored to use international test sets and promising algorithms from the research literature. Furthermore, one of the canonical measures of similarity from the field of information theory, the Kullback relative information, was also implemented and compared to the metrics based on maximum likelihood.

In general, image retrieval by content requires algorithms for extracting and comparing features. Extracted features from the imagery may be associated with entire digital images, or perhaps with specific regions of interest that are identified interactively, semi-automatically, or in a completely automatic manner. The QBIC effort is one project that has developed several methods for doing this. For example, the authors [7] represent the texture in an image by a feature vector and compute the distance between feature vectors using the SSD. Retrieval of similar images is accomplished by finding the N database images which have the shortest distance between feature vectors. Another approach similar to QBIC is described in [28]. This technique matches a pattern against equal-sized identically-oriented regions of a larger image and applies two criteria that roughly correspond to the color and texture criteria of QBIC.

The authors consider the difference between the pattern and the image in a particular relative position as being the SSD between the pattern and the intensity image.

Color indexing is one of the most prevalent retrieval methods in content based image retrieval. Given a query image, the goal is to retrieve all the images whose color compositions are similar to the color composition of the query image. Typically, the color content is described using a histogram [29]. In general, color histograms are computed and the histogram intersection criterion is used to compare them. In [26], efficient techniques for comparing histograms using quadratic measures of similarity have been proposed. Hafner, et al. [11] suggest the usage of a more sophisticated quadratic form of distance measure which tries to capture the perceptual similarity between any two colors. In all of these works, most of the attention has been focussed on the color model with little or no consideration of the noise models.

A method for calculating the similarity between two digital images using a global signature which includes texture, shape, and color content is described in [17] and [19]. A normalized distance between probability density functions of feature vectors is used to match signatures. The authors present four possible distance measures that can be used to compare signatures, without discussing how each of these distances influences the retrieval results.

Stereo matching implies finding correspondences between two or more images. If these correspondences can be found accurately and the camera geometry is known, then a 3D model of the environment can be reconstructed [23], [2]. Several algorithms have been developed to compute the disparity between images, e.g. the correlation methods [22] or correspondence methods [10]. In [8], pixel correspondences are found by adaptive, multi-window template matching. The templates are compared using the SSD. Recent research by [3] concluded that the SSD is sensitive to outliers and therefore robust M-estimators should be used for stereo matching. However, the authors [3] did not consider metrics based on similarity distributions. They considered ordinal metrics, where an ordinal metric is based on relative ordering of intensity values in windows - rank permutations. Cox, et al. [6] presented a stereo algorithm that optimizes a maximum likelihood cost function. This function assumes that corresponding features in the left and right images are normally distributed about a common true value. However, the authors [6] noticed that the normal distribution assumption used to compare corresponding intensity values is violated for some of their test sets. They altered the stereo pair so that the noise distribution would be closer to a Gaussian. In our approach, we attempt to find a better model for the real noise distribution instead of altering the stereo pair.

Boie and Cox [5] consider a model of camera noise comprised of stationary direction-dependent electronic noises combined with fluctuations due to signal statistics. These fluctuations enter as a multiplicative noise and are non-stationary and vary over the scene. A substantial simplification appears if the noise can be modeled as Gaussian distributed and stationary. This work is complementary to ours. They try to model the imaging noise. We try to model the noise between two images which are different due to differing handling and storage conditions of the original photographs, varying orientation, motion, or printer noise.

Section 2 describes the mathematical support for the maximum likelihood approach. The setup of our experiments is given in Section 3. In Section 4 we apply the theoretical results from Section 2 to determine the influence of the real noise model on the accuracy of retrieval methods in image databases. In Section 5 we study the real noise model to be chosen in stereo matching applications. The same approach is then applied on a video sequence in Section 6. Conclusions are given in Section 7.

2 Maximum Likelihood Approach

Maximum likelihood theory [14] [12] [25] allows us to relate a noise distribution to a metric. Specifically, if we are given the noise distribution then the metric which maximizes the similarity probability [27] is

$$\sum_{i=1}^{M} \rho(n_i) \tag{1}$$

where n_i represents the ith bin of the discretized noise distribution and ρ is the maximum likelihood estimate of the negative logarithm of the probability density of the noise. In practice, the noise distribution is typically represented by the difference between the corresponding elements given by the ground truth.

To analyze the behavior of the estimate we take the approach described in [12] and [25] based on *influence function.* The influence function characterizes the bias that a particular measurement has on the solution and is proportional to the derivative, ψ, of the estimate [4]:

$$\psi(z) \equiv \frac{d\rho(z)}{dz} \tag{2}$$

In case the noise is Gaussian distributed:

$$Prob\{n_i\} \sim \exp(-{n_i}^2) \tag{3}$$

then

$$\rho(z) = z^2 \qquad \psi(z) = z \tag{4}$$

If the errors are distributed as a *double* or *two-sided exponential*, namely

$$Prob\{n_i\} \sim \exp(-|n_i|) \tag{5}$$

then, by contrast,

$$\rho(z) = |z| \qquad \psi(z) = sgn(z) \tag{6}$$

In this case, using equation (1), we minimize the *mean absolute deviation*, rather than the *mean square deviation.* Here the tails of the distribution, although exponentially decreasing, are asymptotically much larger than any corresponding Gaussian.

A distribution with even more extensive - therefore sometimes even more realistic - tails is the *Cauchy* distribution,

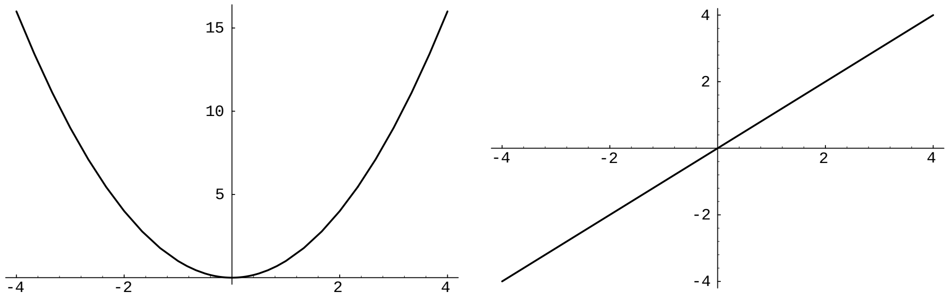
$$Prob\{n_i\} \sim \frac{1}{a^2 + {n_i}^2} \tag{7}$$

where a is a parameter which determines the height and the tails of the distribution.

This implies

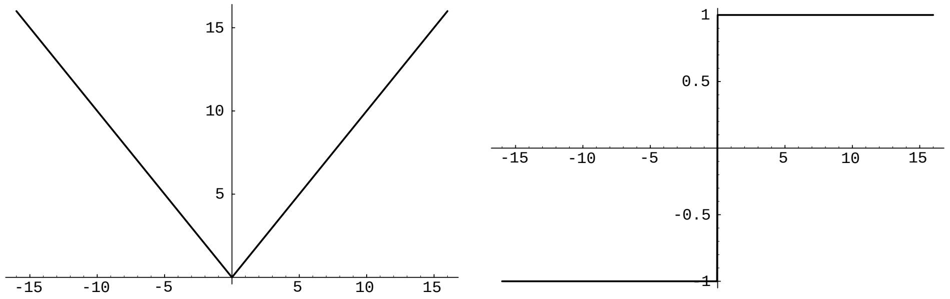
$$\rho(z) = \log(a^2 + z^2) \qquad \psi(z) = \frac{z}{a^2 + z^2} \tag{8}$$

For normally distributed errors, equation (4) says that the more deviant the points, the greater the weight (Figure 1). By contrast, when tails are somewhat more prominent, as in (5), then (6) says that all deviant points get the same relative weight, with only the sign information used (Figure 2). Finally, when the tail are even larger, (8) says that ψ increases with deviation, then starts decreasing, so that very deviant points - the true outliers - are not counted at all (Figure 3).



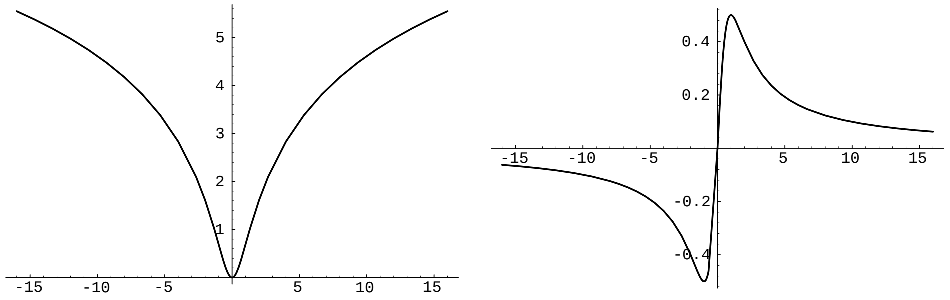
(a) (b)

Figure 1: Quadratic estimator. (a) Estimate, (b) ψ-function



(a) (b)

Figure 2: Exponential estimator. (a) Estimate, (b) ψ-function



(a) (b)

Figure 3: Cauchy estimator. (a) Estimate, (b) ψ-function

The estimates are plotted along with their ψ-functions in Figures 1, 2 and 3.

The additive noise model is the dominant model used in computer vision regarding maximum likelihood estimate. Haralick and Shapiro [13] consider this model in defining the M-estimate: "any estimate T_k defined by a minimization problem of the form $\min \sum_i \rho(x_i - T_k)$ is called an M-estimate". Note that the operation "-" between the estimate and the real data implies an additive model.

In summation, one can note that equation (4) resembles the L_2 metric, while equations (6) and (8) resemble the L_1 and L_c metrics, respectively. Thus, *maximum likelihood* gives a direct connection between the noise distribution and the comparison metrics. Considering ρ as the negative logarithm of the probability density of the noise then, the corresponding metric is given by equation (1).

3 Experimental Setup

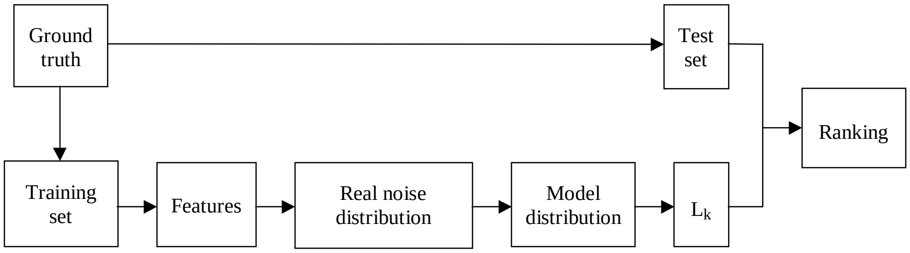


Figure 4: An overview of a similarity matching algorithm

The setup of our experiments is the following. First, we assume that representative ground truth is provided. The ground truth is split into two non-overlapping sets: the training set and the test set, as shown in Figure 4. Note that L_k is a notation for all possible metrics that can be used, e.g. L_1, L_2, L_c. Second, the training set is converted to a histogram which is then normalized to what we denote the real noise distribution. The Gaussian, Exponential, and Cauchy distributions are fitted to the real distribution. The Chi-square test is used to find the fit between each of the model distributions and the real distribution. We select the model distribution which has the best fit and its corresponding metric (L_k) is used in ranking. The ranking is done using only the test set.

For benchmarking purposes we also investigate the performance of other distance measures in matching. In all of the experiments we compare our results with the ones obtained using the Kullback relative information (K) [20]. Let u and v be two discrete distributions then

$$K = \sum_i u_i \log \frac{u_i}{v_i} \tag{9}$$

where the sum is over all bins.

Note that the Kullback relative information is an asymmetric similarity measure between normalized probability density functions. In content based retrieval where normalized histograms are used as feature vectors, K was computed using (9) where u was the feature vector corresponding to the query and v was the feature vector corresponding to a candidate match. In stereo matching and motion tracking where template matching is performed, suppose we are searching for a match for an intensity vector U from the left image. In the right image there will be many possible matching vectors and let V be one of them. Each of the intensity vectors are normalized to have the sum equal to 1 by dividing each component by the total intensity within the vector, i.e. $u_i = U_i / \sum_i U_i$. This results in two normalized vectors u and v and (9) can be applied for computing K.

We chose the Kullback relative information as a benchmark because it is the most frequently used information theoretic similarity measure. Furthermore, Rissanen [24] showed that it serves as the foundation for other minimum description length measures such as the Akaike's [1] information criterion. Regarding the relationship between the Kullback relative information and the maximum likelihood approach, Akaike [1] showed that maximizing the expected log likelihood ratio in maximum likelihood estimation is equivalent to maximizing the Kullback relative information. Another interesting aspect of using the Kullback relative information as a benchmark is that it gives an example of using a logarithmically weighted function, instead of u-v it is computing a weighted version of $\log u - \log v = \log(u/v)$.

It is important to note that for real applications, the parameter in the Cauchy distribution is found when fitting this distribution to the real distribution. This parameter setting would be used for the test set and any future comparisons in that application. The parameter setting can be generalized beyond the ground truth if the ground truth is representative.

For our image retrieval experiments we considered the applications of image retrieval in a black&white image database, printer-scanner copy location, and object recognition by color invariance. In the first experiment, the images have varying kinds of degradation due to different storage conditions, scratches, and writings on the images. In the printer-scanner application, an image is printed to paper and then scanned back into the computer. This task involves noise due to the dithering patterns of the printer and scanner noise. In object recognition, multiple pictures are taken of a single object at different orientations. Therefore, the correct match for an image is known by the creator of the ground truth.

In stereo matching and motion tracking, the ground truth is typically generated manually. A set of reference points are defined in the images and then a person finds the correspondences for the stereo pair or video sequence.

In summary, our algorithm can be described as follows:

Step 1 Compute the feature vectors from the training set

Step 2 Compute the real noise distribution from the differences between corresponding elements of the feature vectors

Step 3 Compare each of the model distributions $\mathcal{M}$ to the real noise distribution $\mathcal{R}$ using the Chi-square test

$$\chi^2 = \sum_i \frac{(\mathcal{R}_i - \mathcal{M}_i)^2}{\mathcal{M}_i} \tag{10}$$

where the sum is over all bins.

Step 3.1 For a parameterized metric such as L_c compute the value a of the parameter that minimizes the Chi-square test

Step 4 Select the corresponding L_k of the best fit model distribution

Step 4.1 Use the value a found from **Step 3.1** in the parameterized metrics

Step 5 Apply the L_k metric in ranking

4 Similarity Noise in Image Databases

The image retrieval problem is the following: Let $\mathcal{D}$ be an image database and $\mathcal{Q}$ be the query image. Obtain a permutation of the images in $\mathcal{D}$ based on $\mathcal{Q}$, i.e assign rank$(\mathcal{I}) \in [|\mathcal{D}|]$ for each $\mathcal{I} \in \mathcal{D}$, using some notion of similarity to $\mathcal{Q}$. The problem is usually solved by sorting the images $\mathcal{Q}' \in \mathcal{D}$ according to $|f(\mathcal{Q}')-f(\mathcal{Q})|$, where $f(\cdot)$ is a function computing feature vectors of images and $|\cdot|$ is some distance measure defined on feature vectors.

One of the problems with query information retrieval systems is that the result of a query is simply a group of items that are hopefully interesting to the user (a group of images that are similar to the query image). Some additional information, such as similarity scores produced by the comparison process, might also be returned to allow a user to gauge the correctness of the result. It is therefore reasonable for a user to pose a question such as, "Why do these images look similar ?" Using a probability density function approach one can give an objective answer to this question [18].

We applied the theoretical results described in Section 2 in two experiments. First, we determined the influence of the similarity noise model on the similar image retrieval performance in a black&white image database: the Leiden 19^{th} Century Portrait Database (LCPD). Second, in order to have a broader range of test data, we used two color image databases. The first one was the Corel Photo database and the second one consisted of 500 reference images of domestic objects, tools, toys, food cans, art artifacts, etc.

4.1 Experiments Using LCPD

The LCPD is currently composed of 16,384 images taken during the 19^{th} century and will be continually expanded until at least 50,000 images are in the database. Some images are copies of each other. However, due to different storage conditions, the copies have varying kinds and differing amounts of degradation. The degradation varies from intensity and moisture damage to scratches and writing on the images as shown in Figure 5.

Our ground truth consisted of 292 copy pairs. We used 100 image pairs from the ground truth as the training set and then calculated the real noise distribution as the normalized histogram of differences between corresponding image elements. In the next step, we compared the real distribution with each of the known distributions: Gaussian, Exponential, and Cauchy. Furthermore, for each of the 192 copy pairs in the test set, we queried the database using the corresponding metric and inspected how it affected the retrieval results.

For comparing the retrieval results we used the performance measures given in [16]. We

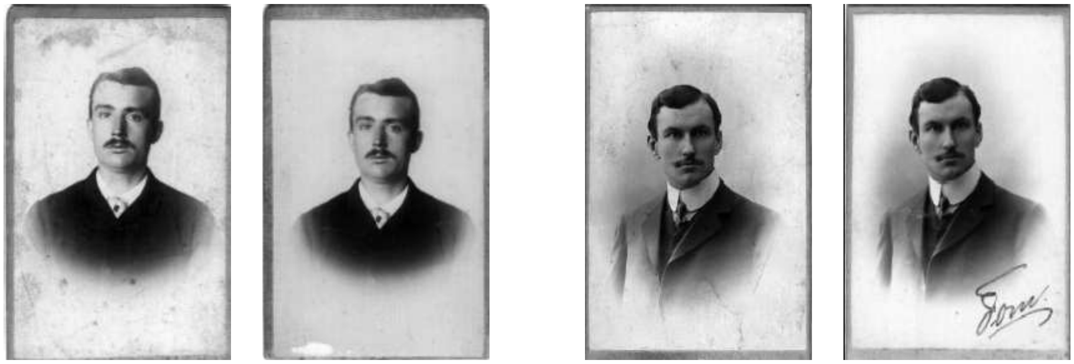


Figure 5: Two examples of copy pairs from LCPD

wanted the performance measures to be some function of the database size. Therefore, we chose a visible window size of length $L = [\log_2 n]$, with $n =$ database size, which ensures a reasonable number of images displayed to the user. This means that for our present database size of 16,384 images, the number of displayed images was 14. For a database consisting of 1 million images, no more than 20 images would have to be shown.

Let T represent the total number of test pairs and T_v be the number of copies which appear in the top $L = [\log_2 n]$ ranks. The **visible fraction** (F_v) is defined as the fraction of correct copies seen by the user,

$$F_v = T_v/T \tag{11}$$

and is normalized to lie within [0,1]. F_v indicates how often copies can be found in the first view shown after a search has been specified.

A second performance measure is the **visible position** (P_v) which is defined as the ranking accuracy within the display window.

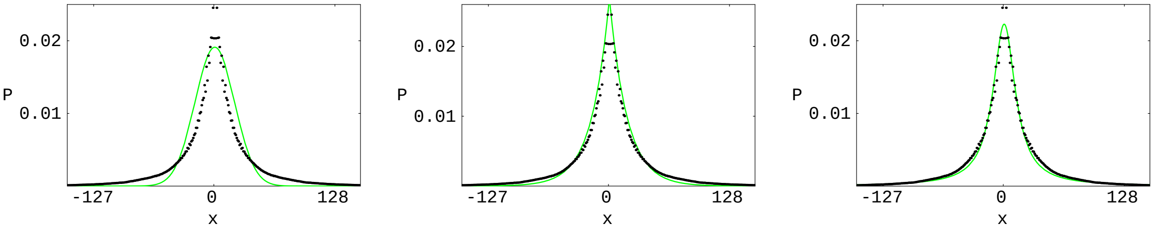
$$P_v = (L - R_v)/(L - 1) \tag{12}$$

where R_v is the average rank for visible test-pairs. P_v lies within [0,1]: 0 when $R_v = L$ and is 1 when $R_v = 1$ (all visible test-pairs on top). P_v acts as a fine tuning measure within the display window. This measure is mainly used to discriminate between methods that have the same number of test-pairs visible (they have the same F_v). Consider for example that two methods have all test-pairs visible (F_v=1) but one has the average rank in the display window (R_v) smaller than the other, meaning that its P_v is greater. In this case, P_v indicates that this method performs better than the other.

Finally as a global measure we used the combined **retrieval quality** Q_r:

$$Q_r = P_v * F_v \tag{13}$$

In the LCPD experiments, we used the projection features introduced in [15]. This feature proved to be one of the best features for copy location. We used average row- and column intensity values (line integrals) as a feature vector.



(a) Gauss (b) Exponential (c) Cauchy (a=14.20)

Figure 6: Real noise distribution in feature space modeled by three theoretical distributions (the approximation error is: (a) 0.031; (b) 0.015; (c) 0.013)

In Figure 6 we displayed the real noise distribution (with dots) along with the three distributions. The approximation error between the real noise distribution and each of the known distributions was calculated using a Chi-square test.

The tails of the real distribution are prominent, so the Gaussian distribution cannot be a good match. Instead, the Exponential and Cauchy distributions are more suitable as approximations. These observations are in accordance with the theory described in Section 2. Therefore, one expects to obtain better overall retrieval results using L_c or L_1 than using L_2, which is corroborated by the experiments in Table 1. The retrieval quality obtained with L_1 and L_c is significantly greater than the one obtained with L_2. Note that the Kullback relative information performs better than L_2 and L_1, but worse than L_c.

The influence of the parameter a in the retrieval quality is shown in Figure 7. For a wide

Methods	Proj			
	L_2	L_1	L_c (a=14.2)	K
F_v	0.842	0.865	0.876	0.869
P_v	0.875	0.879	0.881	0.879
Q_r	0.737	0.761	0.772	0.764

Table 1: Similar image retrieval performance in LCPD

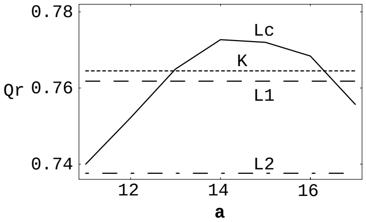


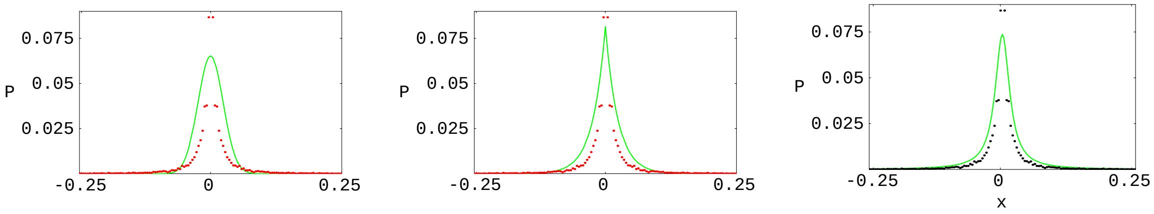
Figure 7: Retrieval quality in LCPD

scale of values for a the results using L_c are better than the ones using L_2. Furthermore, around the optimum value of the parameter the results are better than the ones obtained using L_1 or K. It should be noted that our method for finding the parameter a is only effective when representative ground truth is available.

4.2 Experiments with Color Databases

The first experiments were done using 11,000 images from the Corel database. We used this database because it represents a widely used set of photos by both amateur and professional graphical designers. Furthermore, it is available on the Web at http://www.corel.com.

Before we can measure the accuracy of particular methods, we first had to find a challenging and objective ground truth for our tests. The idea of our experiments was to measure the effectiveness of a retrieval method when trying to find a copy of an image in a magazine or newspaper. In order to create the ground truth we printed 110 images using an Epson Stylus 800 color printer at 720 dots per inch and then scanned each of them at 400 pixels per inch using an HP IIci color scanner. Note that we purposely chose a hard test set. The query image is typically very different from the target image. The copy pairs typically differ by color shifts, quantization artifacts, and dithering noise.



(a) Gauss (b) Exponential (c) Cauchy (a=4.32)

Figure 8: Noise distribution in Corel database compared with the best fit Gaussian (a) (approximation error is 0.106), best fit Exponential (b) (approximation error is 0.082) and best fit Cauchy (c) (approximation error is 0.068)

We used the HSV color model and quantized H using 4 bits, S using 2 bits, and V using 2 bits. The first question we asked was, "Which distribution is a good approximation for the real color model noise?" To answer this we needed to measure the noise with respect to the color model. The real noise distribution was obtained as the normalized histogram of differences between the elements of color histograms corresponding to copy-pair images from the training set (50 image pairs).

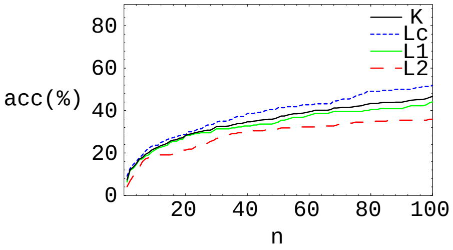


Figure 9: Retrieval accuracy in Corel database for the top 100; for L_c, a=4.32

The best fit Exponential had a better fit to the noise distribution than the Gaussian (Figure 8). Consequently, this implies that L_1 should have better retrieval accuracy than L_2. The Cauchy distribution is the best fit overall, and the results obtained with L_c reflect this. For the retrieval accuracy we chose to display the percentage of correct copies found within the top n matches. From the tests, as shown in Figure 9, it is clear that L_c gives a significant improvement in retrieval accuracy as compared to L_2 and L_1. The Kullback relative information gives slightly better results than L_2 or L_1. Note that we could have simplified the test by reducing the size of the database from 11,000 images to 1,100 images, but then the differences between the distance measures might not have been apparent.

In the second experiment we used a database consisting of 500 images of domestic objects, tools, toys, food cans, etc. As ground truth we used 48 images of 8 objects taken from different camera viewpoints (6 images for a single object). For this experiment we chose to implement a method designed for indexing by color invariants. Our goal was to study the influence of the similarity noise on the retrieval results.

Gevers, et al. [9] analyzed and evaluated various color features for the purpose of image retrieval by color-metric histogram matching under varying illumination environments. They introduced a new color model l and showed that it is invariant for both matte and shiny surfaces:

$$l_1(R,G,B) = \frac{(R-G)^2}{(R-G)^2+(R-B)^2+(G-B)^2} \tag{14}$$

$$l_2(R,G,B) = \frac{(R-B)^2}{(R-G)^2+(R-B)^2+(G-B)^2} \tag{15}$$

$$l_3(R,G,B) = \frac{(G-B)^2}{(R-G)^2+(R-B)^2+(G-B)^2} \tag{16}$$

where R, G, B are the color values in the RGB color space.

The authors [9] concluded that this color model is the most appropriate color model to be used for image retrieval by color-metric histogram matching under the constraint of a white illumination source. This conclusion was drawn using histogram intersection (L_1) as the comparison metric between the color histograms.

Using 24 images with varying viewpoint as the training set, we calculated the real noise distribution and studied the influence of different distance measures on the retrieval results. We used the l color model introduced before and we quantized each color component with 3 bits resulting in color histograms with 512 bins. The problem is formulated as follows: Let $\mathcal{Q}_1, \cdots, \mathcal{Q}_n$ be the query images and for the i-th query $\mathcal{Q}_i$, $\mathcal{I}_1^{(i)}, \cdots, \mathcal{I}_m^{(i)}$ be the images similar with $\mathcal{Q}_i$ according to the ground truth. The retrieval method will return this set of answers with various ranks. As an evaluation measure of the performance of the retrieval method we used recall vs. precision at different scopes: For a query $\mathcal{Q}_i$ and a scope $s > 0$, the recall r is defined as $|\{\mathcal{I}_j^{(i)}|rank(\mathcal{I}_j^{(i)}) \leq s\}|/m$, and the precision p is defined as $|\{\mathcal{I}_j^{(i)}|rank(\mathcal{I}_j^{(i)}) \leq s\}|/s$.

Methods	Precision			Recall		
Scope	5	10	25	5	10	25
L_2	0.425	0.2583	0.1283	0.425	0.5166	0.6416
L_1	0.45	0.2708	0.135	0.45	0.5416	0.675
K	0.466	0.2791	0.1383	0.466	0.5583	0.6916
L_c (a=7.5)	0.525	0.2958	0.146	0.525	0.5916	0.733

Table 2: Recall/Precision vs Scope

The Cauchy distribution was the best match for the measured noise distribution. The Exponential distribution was a better match than the Gaussian. Table 2 shows the precision and recall values at various scopes. The results obtained with L_c were consistently better than the ones obtained with the other measures.

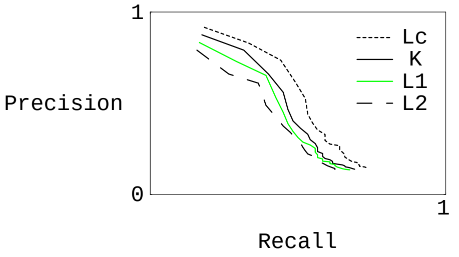


Figure 10: Precision/Recall for color objects database; for L_c, a=4.32

Figure 10 shows the precision-recall graphs. The curve corresponding to L_c is above the others showing that the method using L_c is more effective. Note that the Kullback relative information performs better than L_1 or L_2.

In summary, L_c performed better than all of the other measures. It is interesting that the Kullback relative information performs consistently better than the well-known histogram intersection (L_1).

5 Similarity Noise in Stereo Matching Applications

Stereo matching is the process of finding correspondences between entities in images with overlapping scene content. The images are typically taken from cameras at different viewpoints which implies that the intensity of corresponding pixels may not be the same.

In the first experiments we used two standard stereo data sets (Castle set and Tower set) provided by Carnegie Mellon University. These datasets contain multiple images of static scenes with accurate information about object locations in 3D. The images were taken with a scientific camera in an indoor setting at the Calibrated Imaging Laboratory at CMU. The 3D locations are given in X-Y-Z coordinates with a simple text description (at best accurate to 0.3 mm) and the corresponding image coordinates (the ground truth) are provided for all eleven images taken for each scene. For each image there are provided 28 points as ground truth in the Castle set and 18 points in the Tower set. An example of two stereo images from the Castle data set is given in Figure 11.

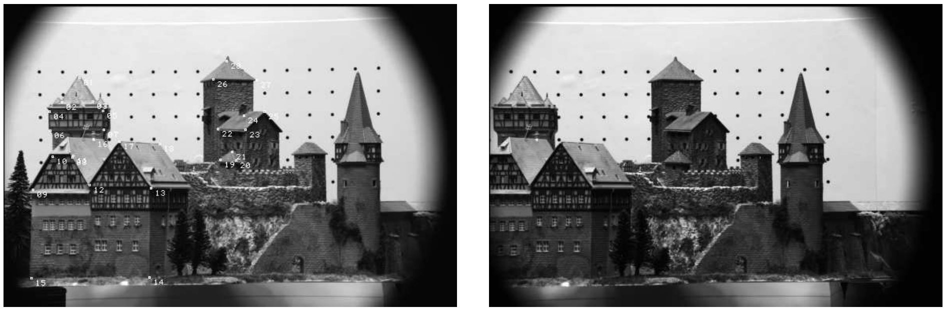


Figure 11: A stereo image pair from the Castle data set

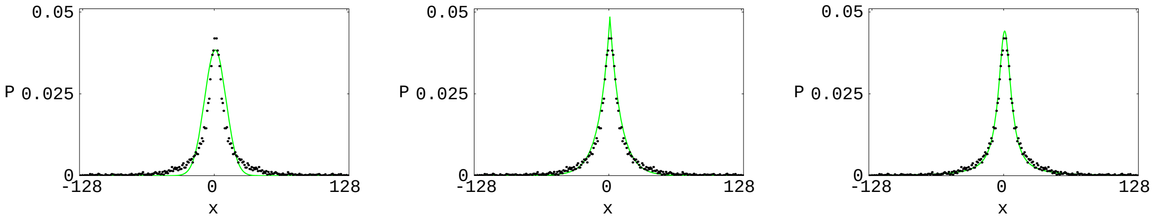
Let $\mathcal{I}_1$ and $\mathcal{I}_2$ represent intensities in two templates i.e. there exist n tuples $(\mathcal{I}_1^1, \mathcal{I}_2^1), \cdots,$ $(\mathcal{I}_1^n, \mathcal{I}_2^n)$, n depending on the size of the template used. The quantity

$$SSD = \sum_{i=1}^{n} (\mathcal{I}_1^i - \mathcal{I}_2^i)^2 \tag{17}$$

measures the squared Euclidean distance (L_2) between $(\mathcal{I}_1, \mathcal{I}_2)$ and a value close to zero indicates a strong match. The other metrics L_1 and L_c can be defined similarly.

In each image we considered the templates around points which were given by the ground truth. We wanted to find the model for the real noise distribution which assured the best accuracy in finding the corresponding templates in the other image. As a measure of performance we computed the accuracy of finding the corresponding points in the neighborhood of one pixel around the points provided by the test set. In searching for the corresponding pixel, we examined a band of height 7 pixels and width equal to the image dimension centered at the row coordinate of the pixel provided by the test set.

In this application we used a template size of n=25, i.e. a 5 × 5 window around the central point. For the training sets, we placed templates around 10 points which were obtained from the ground truth.



(a) Gauss (b) Exponential (c) Cauchy (a=7.47)

Figure 12: Noise distribution in the stereo matcher using Castle dataset

Image set	Gauss	Exponential	Cauchy
Castle	0.0486	0.0286	0.0246
Tower	0.049	0.045	0.043

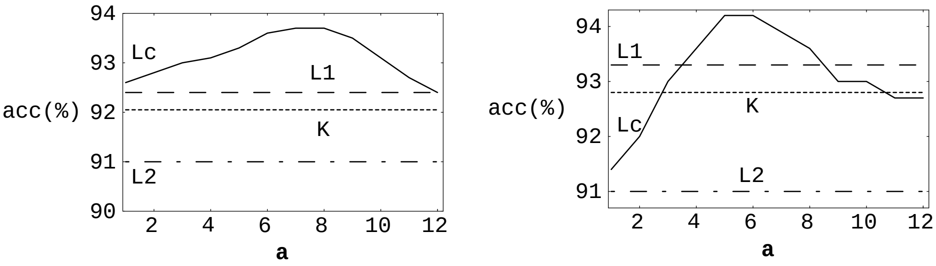
Table 3: The approximation error for the corresponding point noise distribution in stereo matching for three distribution models

We present the real noise distribution in Figure 12. As one can see from Table 3 the Cauchy distribution has the best fit to the measured distribution. Therefore, one expects the accuracy to be the greatest when using L_c (Table 4). In all cases (Figure 13) the results obtained with L_2 are the worst. Furthermore, L_c has the best accuracy relative to the other similarity measures for both test sets.

Image set	L_2	L_1	K	L_c
Castle	91.05	92.43	92.12	93.71 (a=7.47)
Tower	91.11	93.32	92.84	94.26 (a=5.23)

Table 4: The accuracy (%) of the stereo matcher

In addition, we investigated the influence of similarity noise using two promising stereo



(a) Castle dataset (b) Tower dataset

Figure 13: The accuracy of the stereo matcher

algorithms and another stereo pair from the research literature. Our intention was to try other distance measures than SSD (which was used in the original algorithms) in calculating the disparity map.

The first algorithm [8] is an adaptive, multi-window scheme using left-right consistency to compute disparity. For each pixel the correlation with nine different windows (Figure 14) is performed and the disparity with the smallest SSD (L_2) error value is retained. The authors conclude that the adaptive, multi-window scheme clearly outperforms fixed window schemes. Moreover, the left-right consistency check proves to be effective in eliminating false matches and identifying occluded regions.

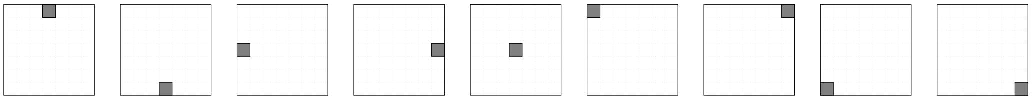


Figure 14: The nine asymmetric correlation windows

The second algorithm we implemented and tested was introduced by Cox, et al. [6]. Their algorithm optimizes a maximum likelihood cost function. This function assumes that corresponding features in the left and right images are normally distributed about a common true value and consists of a weighted squared error term if two features are matched or a (fixed) cost if a feature is determined to be occluded. Their interesting idea was to perform matching on the individual pixel intensity, instead of using an adaptive window as in the area-based correlation methods.

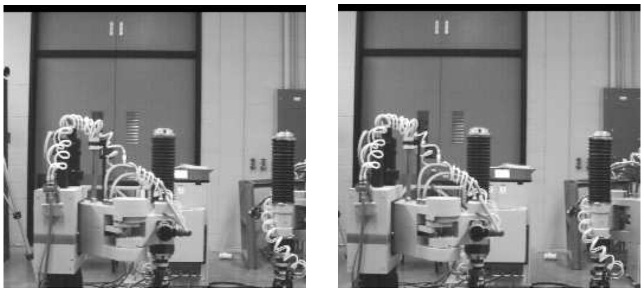
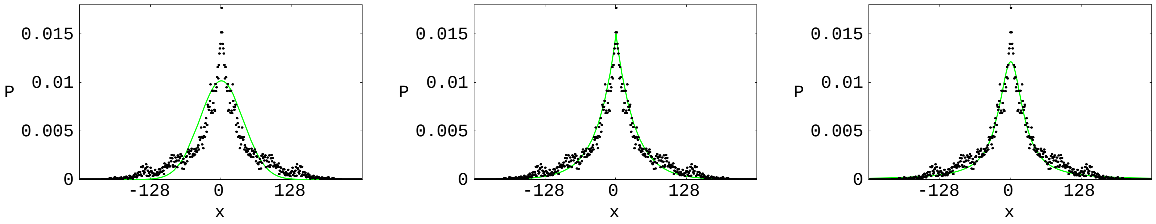


Figure 15: ROBOTS stereo pair

In order to evaluate the performance of the stereo matching algorithms under difficult matching conditions we also used the Robots stereo pair [21]. This stereo pair is more difficult due to varying levels of depth and occlusions (Figure 15). This fact is illustrated in the shape of the real noise distribution (Figure 16). Note that the distribution in this case has wider spread and is less smooth. For this stereo pair, the ground truth consists of 1276 point pairs, given with one pixel accuracy.



(a) Gauss (b) Exponential (c) Cauchy (a=26.26)

Figure 16: Noise distribution for the ROBOTS stereo pair compared with the best fit Gaussian (a) (approximation error is 0.0267), best fit Exponential (b) (approximation error is 0.0156) and best fit Cauchy (c) (approximation error is 0.0147)

Consider a point in the left image given by the ground truth. The displacement of the corresponding point position in the right image is given by the disparity map. The accuracy is given by the percentage of pixels in the test set which are matched correctly by the algorithm.

Figures 17 and 18 show the accuracy of the algorithms when different distance measures were used. Regarding the multiple window algorithm, the usage of L_c provided an improvement in accuracy of about 4% compared with L_1 and 6% compared with L_2. For the algorithm by Cox, et al., using L_c instead of L_2 gave an 8% improvement in accuracy and 6% compared with

L_1. The Kullback relative information had higher accuracy than L_1 and L_2.

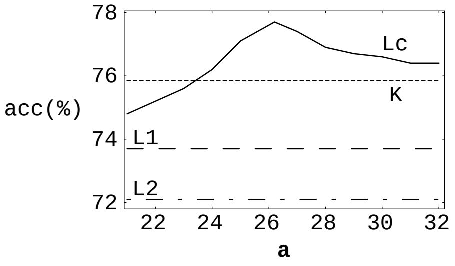


Figure 17: The accuracy of the stereo matcher for the ROBOTS stereo pair using multiple window stereo algorithm

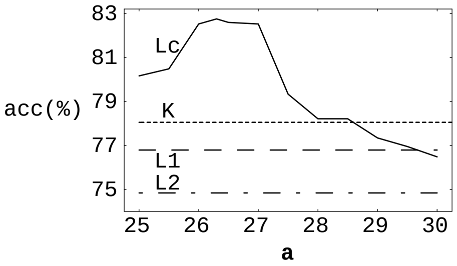


Figure 18: The accuracy of the stereo matcher for the ROBOTS stereo pair using maximum likelihood stereo algorithm

In Tables 5 and 6 the results using different distance measures are presented. For all of the stereo sets L_c had the highest accuracy, and L_2 had the lowest. Note that the accuracy was lower using the ROBOTS stereo pair showing that in this case the matching conditions were more difficult.

Image set	L_2	L_1	K	L_c
Castle	92.27	92.92	92.76	94.82 (a=7.47)
Tower	91.79	93.67	93.14	95.28 (a=5.23)
ROBOTS	72.15	73.74	75.87	77.69 (a=26.2)

Table 5: The accuracy (%) of the stereo matcher using multiple window stereo algorithm

Image set	L_2	L_1	K	L_c
Castle	93.45	94.72	94.53	95.72 (a=7.47)
Tower	93.18	95.07	94.74	96.18 (a=5.23)
ROBOTS	74.81	76.76	78.15	82.51 (a=26.2)

Table 6: The accuracy (%) of the stereo matcher using maximum likelihood stereo algorithm

6 Similarity Noise in Motion Tracking

We used a video sequence containing 19 images on a talking head in a static background [30]. An example of three images from this video sequence is given in Figure 19. For each image in this video sequence there are 14 points given as ground truth. The motion tracking algorithm between the test frame and another frame performed template matching to find the best match in a 5 × 5 template around a central pixel. In searching for the corresponding pixel, we examined a region of width and height of 7 pixels centered at the position of the pixel in the test frame.

The idea of this experiment was to trace moving facial expressions. Therefore, the ground truth points were provided around the lips and eyes which are moving through the sequence. This movement causes the templates around the ground truth points to differ more when far-off frames are considered. This is illustrated in Figure 20.

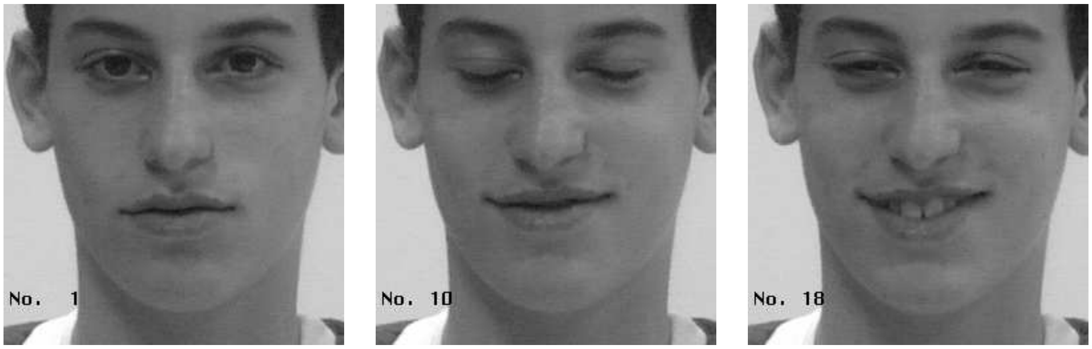


Figure 19: Video sequence of a talking head

Between the first frame and a later frame, the tracking error represents the average displacement (in pixels) between the ground truth and the corresponding pixels found by the matching algorithm. Note that regardless of the frame difference, L_c had the least error and L_2 had the greatest error.

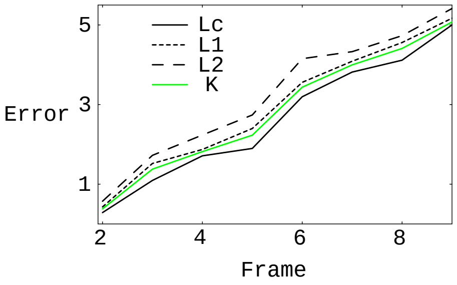
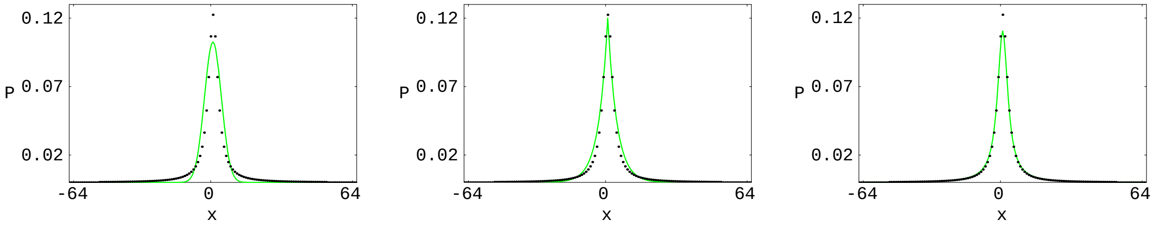


Figure 20: Average tracking error of corresponding points in successive frames; for L_c a=2.03



(a) Gauss

(b) Exponential

(c) Cauchy (a=2.03)

Figure 21: Real noise distribution in the video sequence modeled by three theoretical distributions using sequential frames (the approximation error is: (a) 0.083 ; (b) 0.069 ; (c) 0.063)

In Figure 21 we display the fit between the real noise distribution and the three distributions. The real noise distribution was calculated using templates around points in the training set (6 points for each frame) considering sequential frames. The best fit is the Cauchy distribution, and the Exponential distribution is a better match than the Gaussian distribution. Therefore, it is expected that the accuracy is greater when using L_c than when using L_1 and L_2 (Table 7). For L_c, the greatest accuracy was obtained around the values of the parameter a which gave the best fit between the Cauchy distribution and the real distribution (Figure 22).

In addition, we considered the situation of motion tracking between non-adjacent frames. In Table 7, the results are shown for tracking pixels between frames located at interframe distances of 1, 3, and 5. Note that as the interframe distance increases, the accuracy decreases and the error increases (Figure 20). Overall, L_c gave better results as compared with the other distance measures.

Interframe Distance	L_2	L_1	K	L_c
1	84.11	84.91	85.74	87.43 (a=2.03)
3	74.23	75.36	76.03	78.15 (a=13.45)
5	65.98	67.79	68.56	70.14 (a=21.15)

Table 7: The accuracy (%) of the matching process in video sequence

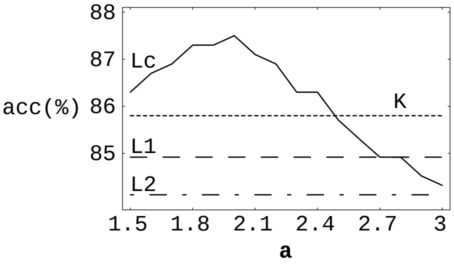


Figure 22: The accuracy of the matching process in video sequence using sequential frames

7 Conclusions and Discussion

In summary, we examined three topic areas from computer vision which were content based retrieval, stereo matching, and motion tracking. Regarding content based retrieval, the first application we examined was finding copies of historical images which had suffered different handling and storage conditions. Previous research had shown that row and column projections were an effective method for copy location. The second application was finding copies of images which had been printed and then scanned. For this application we used the Corel stock photo database and a color histogram method for finding the copies. The third application dealt with object recognition using color invariance. Both the ground truth and the algorithm came from the work by Gevers, et al. [9]. Note that in their work, they used the SAD metric.

The second topic area we examined was stereo matching. We implemented a template matching algorithm, an adaptive, multi-window algorithm by Fusiello [8], and a maximum likelihood method using pixel intensities by Cox, et al. [6]. Note that the SSD was used in the work by Fusiello [8] and in the work by Cox [6].

Motion tracking was the third topic area. In these experiments, we implemented a template matching algorithm to track pixels on a moving object in a video sequence. We examined the tracking error and accuracy between adjacent and non-adjacent frames.

For all of the topic areas and applications in our experiments, better accuracy was obtained when the Cauchy metric was substituted for the SSD, SAD, or Kullback relative information. Minimizing the Cauchy metric is optimal with respect to maximizing the likelihood of the difference between image elements when the real noise distribution is equivalent to a Cauchy distribution. Therefore, the breaking points occur when there is no ground truth, the ground truth is not representative or when the real noise distribution is not a Cauchy distribution. We also make the assumption that one can measure the fit between the real distribution and a model distribution, and that the model distribution which has the best fit should be selected. We used the Chi-square test as the measure of fit between the distributions, and found in our experiments that it served as a reliable indicator for distribution selection.

The first problem addressed in this paper is whether the SSD is appropriate to use for computer vision applications in content based retrieval, stereo matching, and motion tracking. From our experiments, the SSD is typically not justified because the real noise distribution is not Gaussian.

There appear to be two methods of applying maximum likelihood toward improving the accuracy of matching algorithms. The first method recommends altering the images so that the measured noise distribution is closer to the Gaussian and then using the SSD. The second method is to find a metric which has a distribution close to the real noise distribution. Our experiments suggest that real noise distributions can be modeled using the Cauchy distribution better than with the Gaussian or Exponential. Furthermore, the Kullback relative information also appears to be more accurate in our experiments than the SSD, but not as accurate as the Cauchy metric. Either method has the potential to improve the accuracy of a wide range of vision algorithms (such as content-based retrieval, stereo matching, and motion tracking).

Therefore, our main contributions are in showing that the prevalent Gaussian distribution assumption is often invalid, and in proposing the Cauchy metric as an alternative to both the SAD and Kullback relative information. Furthermore, in the case where representative ground truth can be obtained for an application, we provide a method for selecting the appropriate metric. Overall, it is our recommendation that one should determine whether the model distribution fits the real distribution before using the metric.

In future work we intend to examine the influence of multi-parameter distributions towards achieving a better fit to the real distribution.

References

[1] H. Akaike. Information theory and an extension of the maximum likelihood principle. *2nd International Symposium on Information Theory*, pages 267–281, 1973.

[2] S. Barnard and M. Fischler. Computational stereo. *ACM Computing Surveys*, 14(4):553–572, 1982.

[3] D. N. Bhat and S. K. Nayar. Ordinal measures for image correspondence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(4):415–423, 1998.

[4] M. J. Black. *Robust Incremental Optical Flow.* PhD thesis, Yale University, September 1992.

[5] R. Boie and I. Cox. An analysis of camera noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(6):671–674, 1992.

[6] I. Cox, S. Hingorani, and S. Rao. A maximum likelihood stereo algorithm. *Computer Vision and Image Understanding*, 63(3):542–567, 1996.

[7] M. Flicker, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by image and video content: The QBIC system. *IEEE Computer*, 28(9):23–32, 1995.

[8] A. Fusiello, V. Roberto, and E. Trucco. Efficient stereo with multiple windowing. *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 858–863, 1997.

[9] T. Gevers and A. Smeulders. Color-based object recognition. *Pattern Recognition*, 32(3):453–464, 1999.

[10] W. Grimson. Computational experiments with a feature based stereo algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(1):17–34, 1985.

[11] J. Hafner, H. S. Sawhney, W. Equitz, M. Flickner, and W. Niblack. Efficient color histogram indexing for quadratic form distance functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(7):729–736, 1995.

[12] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel. *Robust Statistic: The Approach Based on Influence Functions.* John Wiley and Sons, New York, 1986.

[13] R. Haralick and L. Shapiro. *Computer and Robot Vision II.* Addison-Wesley, 1993.

[14] P. J. Huber. *Robust Statistic.* John Wiley and Sons, New York, 1981.

[15] D. P. Huijsmans and M. S. Lew. Efficient content-based image retrieval in digital picture collections using projections:(near)copy locations. In *Proc. of 13th International Conference on Pattern Recognition*, volume 3, pages 104–108, 1996.

[16] D. P. Huijsmans, M. S. Lew, and D. Denteneer. Quality measures for interactive image retrieval with a performance evaluation of two 3x3 texel-based methods. In *Lectures Notes in Computer Science*, volume 1311(2), pages 22–29. Springer-Verlag, 1997.

[17] P. M. Kelly and T. M. Cannon. CANDID: Comparison algorithm for navigating digital image databases. *Proc. of the 17th International Working Conference on Scientific and Statistical Database Management*, pages 252–258, 1994.

[18] P. M. Kelly, T. M. Cannon, and J. E. Barros. Efficiency issues related to probability density function comparison. *SPIE - Storage and Retrieval for Image and Video Databases*, 2670(4):42–49, 1996.

[19] P. M. Kelly, T. M. Cannon, and D. R. Hush. Query by image example: the CANDID approach. *SPIE - Storage and Retrieval for Image and Video Databases*, 2420(3):238–248, 1995.

[20] S. Kullback. *Information theory and statistics.* Dover Publications, 1968.

[21] M. S. Lew, T. S. Huang, and K. Wong. Learning and feature selection in stereo matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):869–882, 1994.

[22] W. Luo and H. Maitre. Using surface model to correct and fit disparity data in stereo vision. *International Conference on Pattern Recognition*, 1:60–64, 1990.

[23] D. Marr and T. Poggio. A computational theory of human stereo vision. *Proc. Royal Society Lond.*, 204:301–328, 1976.

[24] J. Rissanen. Modeling by shortest data description. *Automatica*, 14:465–471, 1978.

[25] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection.* John Wiley and Sons, New York, 1987.

[26] H. S. Sawhney and J. L. Hafner. Efficient color histogram indexing. In *Proc. of 1994 IEEE International Conference on Image Processing*, volume 2, pages 66–70, 1994.

[27] N. Sebe, M. S. Lew, and D. P. Huijsmans. Which ranking metric is optimal? with applications in image retrieval and stereo matching. *International Conference on Pattern Recognition*, pages 265–271, 1998.

[28] H. S. Stone and C. S. Li. Image matching by means of intensity and texture matching in the Fourier domain. *SPIE - Electronic Imaging: Science and Technology*, January 1996.

[29] M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[30] L. Tang, Y. Kong, L. S. Chen, C. R. Lansing, and T. S. Huang. Performance evaluation of a facial feature tracking algorithm. *Proceedings of the NSF/ARPA Workshop: Performance vs. Methodology in Computer Vision*, pages 218–229, 1994.

Nicu Sebe received the B.S. and the M.S. Degrees in Electrical Engineering from ”Politechnica” University of Bucharest, Romania, in 1995 and 1996, respectively. Since 1997, he is with the Leiden Institute of Advanced Computer Science, The Netherlands, and is currently writing his doctoral dissertation. His main interest is in the fields of Computer Vision and Pattern Recognition, in particular content-based retrieval and maximum likelihood analysis.

Michael S. Lew received his Ph. D. in Electrical Engineering from the University of Illinois at Urbana-Champaign in 1995. Currently he is an assistant professor and academic fellow of the Leiden Institute of Advanced Computer Science (LIACS) at Leiden University in the Netherlands. In 1995, he had the best research proposal from the Dutch National Science Foundation, and in 2000, he was ranked as the top young scientific researcher from the faculty of science at Leiden University. He has published over 70 articles in refereed journals and conferences in the fields of multimedia search, computer vision, human computer interaction (HCI), and virtual communities. In addition to research and teaching, he is the Co-Director of the LIACS Media Lab.

Dionysius P. Huijsmans received his Ph. D. in Mathematics and Physics from the University of Amsterdam in 1982. From 1982 till 1985 he did postdoctoral research aimed at three-dimensional reconstruction from serial sections at the Laboratory for Medical Physics in Amsterdam: an interest area in which he remains active till today. From 1985 onwards he is connected to Leiden University in the Netherlands as assistant professor at the LIACS and ASCI research school. His main research area is computer imagery and his efforts are presently concentrated on developing and evaluating tools for visual content-based searches in very large image databases.