Which ranking metric is optimal ?
with applications in image retrieval and stereo matching

N. Sebe M. Lew D.P. Huijsmans
Department of Computer Science, Leiden University
Postbus 9512, 2300 RA, Leiden, The Netherlands
{nicu mlew huijsman}@wi.leidenuniv.nl

Abstract

Euclidean metric is frequently used in Computer Vision, mostly ad-hoc without any justification. However we have found that other metrics like a double exponential or Cauchy metric provide better results, in accordance with the maximum likelihood approach. In this paper we experiment with different modeling functions for similarity noise and compute the accuracy of different methods using these modeling functions in two kinds of applications: content-based image retrieval from a large database and stereo matching. We provide a way to determine the modeling distribution which fits best the similarity noise distribution according to the ground truth. In the optimum case, when one has chosen the best modeling distribution, its corresponding metric will give the best ranking results for the ground truth provided.

1. Introduction

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 ([6], [1]) is one project that has developed several methods for doing this. As an example, the texture of an image is represented by a feature vector that can be compared to texture feature vectors from other database images using Euclidean distance, thereby allowing the retrieval of images with ”similar” textures.

Another feature vector, color content, is typically described using a histogram. In [8], a histogram of the colors contained in each image is computed, and a L_1 metric is used to compare these color histograms. Also, in [7], efficient techniques for comparing histograms using quadratic measures of similarity have been proposed. A method for calculating the similarity between two digital images using a global signature which includes the texture, shape and color content is described in [4]. A normalized distance between probability density functions of feature vectors is used to match signatures. The authors present four possible distances that can be used to compare signatures without discussing how each of these distances influences the retrieval results.

In [3] the authors compare different retrieval methods, using feature vectors composed of projections, Local Binary Pattern and 2D-Trigrams, and evaluate image indexing and retrieval performance for similarity matches in a large database, as a function of the database size. Similar in [5] different methods for image copy location are compared and evaluated, taking into account their computational efficiency and accuracy with respect to real noise experiments.

In some of these applications it was assumed that the distribution of the similarity noise, defined like pixel-by-pixel difference between two similar images or their corresponding feature elements, is gaussian. In other applications, even when other metrics than L_2 were considered, there were no arguments why these were better.

A general application involving similarity matching might follow the scheme presented in figure 1.

First one has to establish ground truth and extract the feature vectors to be used for comparison. The associated similarity noise distribution can be approximated with a modeling distribution and its associated metric (L_k). In the optimum case, when one has chosen the best modeling distribution, its corresponding metric will give the best ranking results for the ground truth provided.

1.1. Image Retrieval

In image retrieval the problem of finding the optimum modeling distribution for the similarity noise distribution is burdened by the fact that defining a ground truth and determining a testset is difficult. This difficulty is due to the

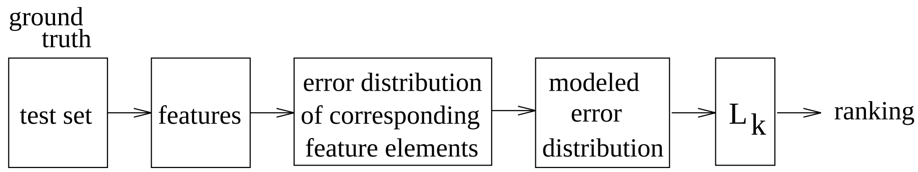


Figure 1. The steps in a similarity matching algorithm

subjectivity of the task: different users may have a completely different idea about the similarity of two images. To have a complete system for evaluation of the performance of content-based similarity matching one would like to have a clear definition of similarity and the right modeling function for similarity noise, resulting in a corresponding metric to base ranking on.

1.2. Stereo Matching

This approach could be applied not only to the applications where the ground truth is defined according to subjectivity but, also in cases where a ground truth is already known. For instance, in stereo datasets the ground truth for matching corresponding points may be provided by the laboratory where these images were taken. This ground truth is a result of mapping the world coordinates, in which the camera is moving, to the image coordinates, using the 3D geometry relations of the scene. In this case one can test automatic stereo matchers that are able to detect the corresponding point pairs registered in stereo images of the testset scenes. For this stereo matcher one can determine the best metric when comparing different image regions to find the similar ones. The optimum metric in this case will give the most accurate stereo matcher.

Section 2 describes the mathematical support for maximum likelihood approach. In Section 3 we apply these theoretical results to determine the influence of the similarity noise model on the accuracy of the retrieval methods in a large database. In Section 4 we study the similarity noise model to be chosen in a stereo matching application. Conclusions are given in Section 5.

2. Maximum likelihood estimator

From the mathematical-statistical point of view, the problem of finding the right model for the similarity noise comes down to the maximization of the similarity probability.

Consider first, two subsets of M images from the database (D) : X ⊂ D, Y ⊂ D which according to the ground truth are similar: $X \equiv Y$

This can be written:

$$x_i \equiv y_i, \qquad i = 1, ..., M \tag{1}$$

where $x_i \in$ X, $y_i \in$ Y represent the images from the corresponding subsets.

The equation (1) can be further written as:

$$x_i = y_i + n_i, \qquad i = 1, ..., M \tag{2}$$

where n_i represent the "noise" image obtained as the difference between the other two images.

In this context the similarity probability can be defined:

$$P(X, Y) = \prod_{i=1}^{M} \{\exp[-\rho(x_i, y_i)]\} \tag{3}$$

where function ρ is the negative logarithm of the probability density of the noise.

According to (3) we have to find the probability density function of the noise that maximizes the similarity probability. This is the *maximum likelihood* estimator for X, given Y ([2]).

We can further suppose that this noise distribution is valid for all the database, so using it for all the images in the database one obtains the best possible ranking results.

In all above considerations, we were talking about images but this notion can be extended to feature vectors associated with the images when we are working with image features or, even, can be extended to pixel values in the images in the case of stereo matching.

Taking the logarithm of (3) we find that we have to minimize the expression:

$$\sum_{i=1}^{M} \rho(x_i, y_i) \tag{4}$$

In this case, according to (2), the function ρ depends not independently on its two arguments, query image x_i and the predicted one y_i, but only on their difference. We have thus a *local* estimator and we can use $\rho(z_i)$ instead of $\rho(x_i, y_i)$, where $z_i \equiv x_i$-y_i and the operation "-" denotes pixel by pixel difference between the images, or an equivalent operation in feature space.

We define the derivative of $\rho(z)$ to be a function $\psi(z) \equiv \frac{d\rho(z)}{dz}$ which will occur as a weighting function when we find the minimum of the expression (4).

In case the noise is gaussian distributed:

$$Prob\{x_i - y_i\} \sim \exp(-[x_i - y_i]^2) \quad (5)$$

then

$$\rho(z) = z^2 \qquad \psi(z) = z \;\; (normal) \quad (6)$$

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

$$Prob\{x_i - y_i\} \sim \exp(-|x_i - y_i|) \quad (7)$$

then, by contrast,

$$\rho(z) = |z| \qquad \psi(z) = sgn(z) \;\; \textit{(double exponential)} \quad (8)$$

In this case the maximum likelihood estimator is obtained by minimizing 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,

$$Prob\{x_i - y_i\} \sim \frac{1}{a^2 + (x_i - y_i)^2} \quad (9)$$

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

This implies

$$\rho(z) = \log(a^2 + z^2) \qquad \psi(z) = \frac{z}{a^2 + z^2} \;\; (Cauchy) \quad (10)$$

For normally distributed errors, equation (6) says that the more deviant the points, the greater the weight. By contrast, when tails are somewhat more prominent, as in (7), then (8) says that all deviant points get the same relative weight, with only the sign information used. Finally, when the tail are even larger, (10) says the ψ increases with deviation, then starts decreasing, so that very deviant points - the true outliers - are not counted at all.

3. Similarity noise in a large image database

We applied the theoretical results described in Section 2 in order to determine the influence of similarity noise model on the similar image retrieval performance in the Leiden 19^{th} Century Portrait Database (LCPD).

The LCPD is currently composed of 6656 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.

For our experiments we used 120 image copy pairs from the database as the ground truth for similarity. According to this ground truth, we determined the real distribution of the similarity noise considering three different spaces: intensity space, gradient and features space. For each of them we tried to match this real distribution best with one of the known distributions: normal, double exponential, Cauchy. We then applied the corresponding similarity noise model to the entire database and inspected how it affected the retrieval results.

For comparing the retrieval results we used the performance measures given in [3]. We consider the **visible fraction** (F_v) that counts how many of the test-pairs **T** have counterparts that appear in the top $L = [\log_2 n]$ ranks: the number of these visible test-pairs are called (T_v) and $n =$ database size. So

$$F_v = T_v / T \quad (11)$$

and is normalized to lie within [0,1].

A second performance measured is the **visible position** (P_v) that is defined like the ranking accuracy within the display window.

$$P_v = (L - R_v)/(L - 1) \quad (12)$$

where R_v is the average rank for visible test-pairs. P_v lies within [0,1]; 0 when $R_v = L$ (all test pairs just visible) and is 1 when $R_v = 1$ (all visible test-pairs on top).

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

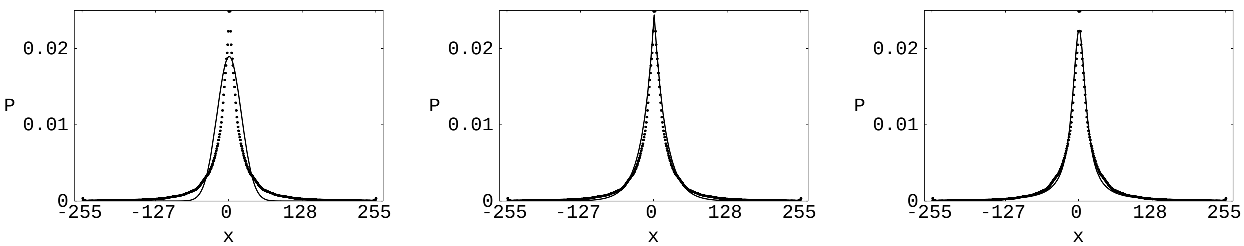
$$Q_r = (F_v + P_v)/2 \quad (13)$$

We will consider that a method provides better results when Q_r is bigger.

3.1. Intensity space

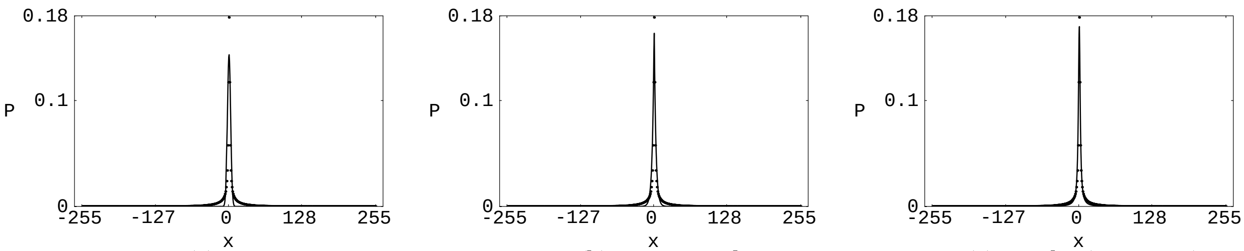
In figure 2 we have the matching of the real similarity noise distribution in the intensity space considering the three distributions. The dotted lines represent the measured noise distribution. As one can notice the tails of the real distribution are prominent so the gaussian distribution cannot be a good match. Instead, the double exponential distribution and also, the Cauchy one are more suitable as approximations. These observations are in accordance with the theory described in the Section 2.

One expects, therefore, to obtain better overall retrieval results using an L_c or L_1 metric than with a ranking based on L_2. As one can see in the first three columns of table 1 the results according to (11), (12) and (13) confirm this supposition.



(a) gauss (b) exponential (c) cauchy (a=14.2057)

Figure 2. Similarity noise distribution for the testset in intensity space modeled by three theoretical distributions (the approximation error is: (a) 0.033 ; (b) 0.013 ; (c) 0.012)



(a) gauss (b) exponential (c) cauchy (a=1.4327)

Figure 3. Similarity noise distribution for the testset in gradient space modeled by three theoretical distributions (the approximation error is: (a) 0.1203 ; (b) 0.0698 ; (c) 0.0459)

But related to the theory, one could ask about the value of the parameter **a** in the Cauchy distribution that should be used to obtain the best results using this distribution.

We consider the following steps to solve this problem:

- perform the matching between the measured noise distribution and the one modeled by a Cauchy distribution and determine the value of the parameter **a** that minimizes the difference between these distributions.

- use this value of parameter **a** in the corresponding metric, L_c.

We were also interested about how the value of this parameter will influence the retrieval quality when we use the Cauchy distribution as a model. This is illustrated in figure 5(a). One can observe that for a wide scale of values for parameter **a** the results using L_c are better that the one using L_2. One can also notice that around the optimum value for the parameter **a** (a $\approx$ 14) the results are the best; even better that the ones obtained using L_1.

3.2. Gradient space

The second space where we examine the influence of the similarity noise model was the gradient space considering the gradient images obtained using a Sobel gradient. We follow the same steps as before.

The measured noise distribution (figure 3) in this case is steeper. So, consequently, the error using the gaussian approximation is even bigger. The retrieval quality measures reflect this (table 1 and figure 5(b)). From the figure 3 we deduce that the results obtained with L_c can be better that the one obtained with L_1, fact illustrated in figure 5(b) around the optimum value for the parameter.

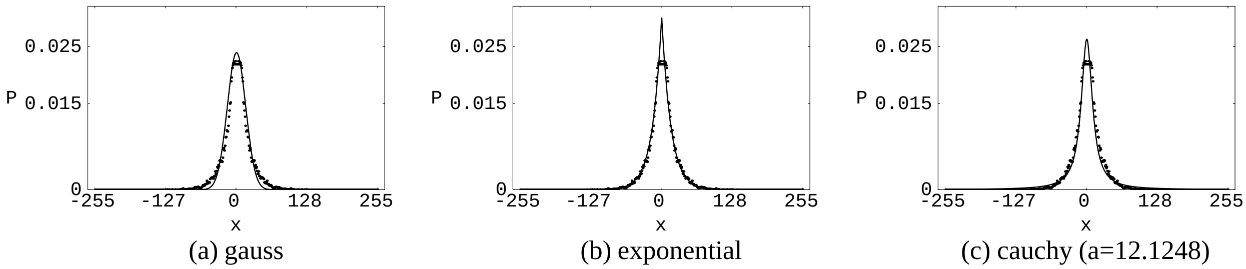


Figure 4. Similarity noise distribution for the testset in feature space modeled by three theoretical distributions (the approximation error is: (a) 0.0306 ; (b) 0.0228 ; (c) 0.0273)

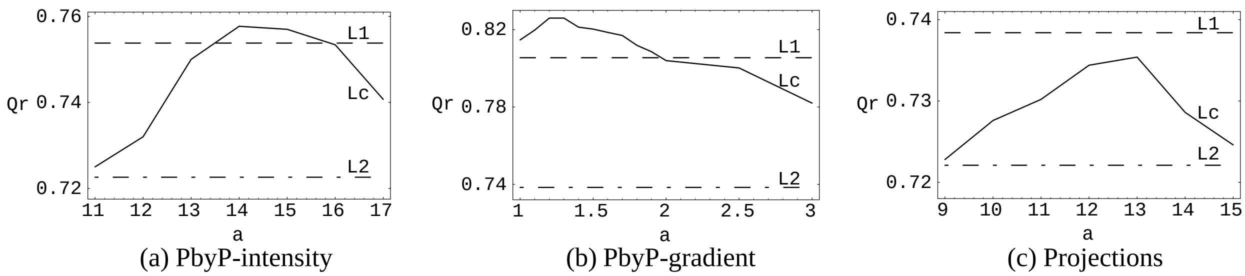


Figure 5. Retrieval quality

3.3. Projection space

In the feature space we consider the projection features from the images. We used average row- and column values (line integrals) as a feature vector. The procedure in this case will use the difference distribution between the corresponding feature elements instead of pixel by pixel difference distribution. In figure 4 one can see that the gaussian distribution in this case more closely matches the measured noise distribution so, the results obtained with L_2 will be close to the ones obtained with the other metrics. Also one can notice that in this case the double exponential distribution matches the measured noise distribution better than the Cauchy one, illustrated consequently in figure 5(c).

4. Similarity noise in stereo matching

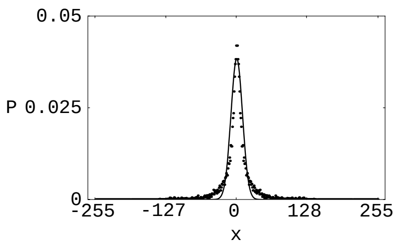
Stereo matching is the process of determining correspondences between entities in related images. One can test automatic stereo matchers that are able to detect these correspondences. The best choice of the optimum metric when compare different image regions in order to find the similar ones will give the most accurate stereo matcher.

We used two standard stereo data sets (Castle set and Tower set) provided by the Calibrated Imaging Laboratory at Carnegie Mellon University. These datasets contain multiple images of static scenes with accurate information about object locations in 3D. 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 Castle set and 9 points in Tower set.

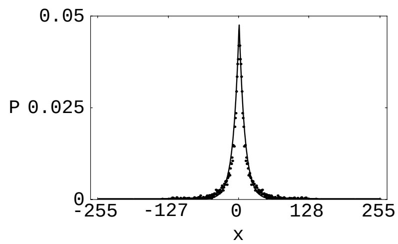
In each of the images we consider the points that are given by the ground truth and we determine the proper similarity noise model that will assure the best accuracy in finding the corresponding points according to the ground truth. We cannot use single pixel information but have to use a region around it. So we will perform template matching. Our automatic stereo matcher will match a template defined around one point from an image with the templates around points in the other images in order to find similar ones. If the

Methods	PbyP-int			PbyP-grad			Proj-int		
	L_2	L_1	L_c	L_2	L_1	L_c	L_2	L_1	L_c
F_v	0.614	0.635	0.650	0.650	0.708	0.750	0.614	0.625	0.620
P_v	0.830	0.872	0.865	0.826	0.903	0.902	0.830	0.851	0.850
Q_r	0.722	0.753	0.757	0.738	0.805	0.826	0.722	0.738	0.735

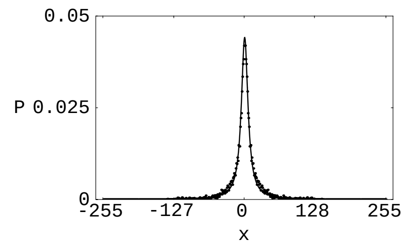
Table 1. Similar image retrieval performance for a test set of 120 near-copies in a database of size 6656 images



(a) gauss



(b) exponential



(c) cauchy (a=7.2007)

Figure 6. Noise distribution in the stereo matcher using Castle dataset

resulting points are equivalent to the corresponding points provided by the ground truth we consider that we have a *hit*, otherwise we have a *miss*. The accuracy is given by the number of the *hits* divided by the number of possible *hits* (number of corresponding point pairs). Because the ground truth is provided with subpixel accuracy we consider the matching to be a *hit* when the corresponding point found lies in the neighborhood of one pixel around the point provided by the ground truth.

Image set	L_2	L_1	L_c
Castle stereo_set	0.0486	0.0286	0.0246
Tower stereo_set	0.0649	0.0475	0.0387

Table 2. The approximation error for the corresponding point noise distribution in stereo matching for three distribution models

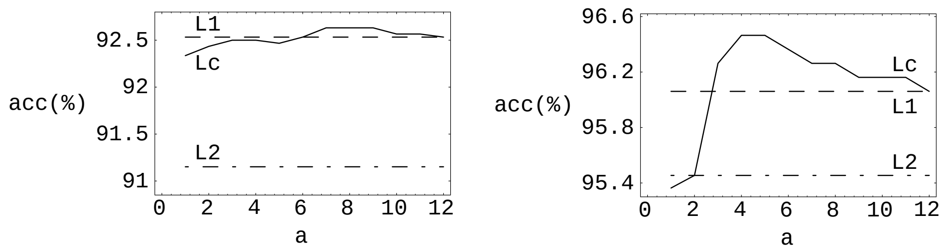
The first step is to compute the similarity noise distribution according to the ground truth. In this case the similarity noise is defined as the difference between two corresponding pixels. The following step is to match this distribution with one of the three distribution and to find out which fits best. We present the corresponding point matching distribution in figure 6. As one can see from the table 2 the Cauchy distribution fits the real distribution best. So one expects the accuracy to be biggest when using L_c (table 3). The best values for the accuracy using L_c are obtained around the optimum value computed for the matching step (figure 7). As one can see from the table 2 the double exponential and Cauchy distribution results are almost identical using the first dataset and deviate a bit more using the second test set; this is illustrated in figure 7.

Image set	gauss	exp	cauchy
Castle stereo_set	91.15	92.53	92.63
Tower stereo_set	95.45	96.06	96.46

Table 3. The accuracy of the stereo matcher

5. Conclusions

In this paper we showed how to derive a suitable metric given a visual query problem when ground truth is available. The final similarity ranking can be improved this way. From our experience L_1 clearly outperform L_2. Using a parametrized metric like L_c one can often further improve the result.



(a) Castle dataset

(b) Tower dataset

Figure 7. The accuracy of the stereo matcher (%)

We showed how maximum likelihood estimation can be successfully applied in content-based query applications.

It was also shown that the behavior of the performance measures given in [3] are in accordance with the maximum likelihood approach.

6. Acknowledgements

This project was supported with a grant from Philips in the Netherlands and by The Netherlands Computer Science Research Foundation as far as M. Lew is concerned.

Data for this research were partially provided by the Calibrated Imaging Laboratory at Carnegie Mellon University, supported by ARPA, NSF, and NASA.

References

[1] C. Faloutsos, M. Flicker, W. Niblack, R. Barder, W. Equitz, and D. Petrovic. Efficient and effective querying by image content. *Journal of Intelligent Information Systems*, pages 231–262, 1994.

[2] P. Huber. *Robust Statistic*. NewYork: Wiley, 1981.

[3] D. Huijsmans, M. 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.

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

[5] M. Lew, D. Huijsmans, and D. Denteneer. Optimal keys for image database indexing. In *Lectures Notes in Computer Science*, volume 1311(2), pages 148–155. Springer-Verlag, 1997.

[6] W. Niblack, R. Barder, W. Equitz, M. Flicker, E. Glasman, D. Petrovic, P. Yanker, C. Faloutsos, and G. Yaublin. The QBIC project: Querying images by content using color, texture and shape. *SPIE - Storage and Retrieval for Image and Video Databases*, 1908:173–181, 1993.

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

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