

UNSUPERVISED THRESHOLDS FOR SHAPE MATCHING

Pablo Musé, Frédéric Sur

Frédéric Cao

Yann Gousseau

CMLA, ENS de Cachan
61, avenue du Président Wilson
94235 Cachan Cedex France

IRISA, INRIA Rennes
Campus Universitaire de Beaulieu
35042 Rennes Cedex France

TSI, ENST
46, rue Barrault
75634 Paris Cedex 13 France

ABSTRACT

Shape recognition systems usually order a fixed number of best matches to each query, but do not address or answer the two following questions: Is a query shape in a given database? How can we be sure that a match is correct? This communication deals with these two key points. A database being given, with each shape S and each distance δ , we associate its *number of false alarms* $NFA(S, \delta)$, namely the expectation of the number of shapes at distance δ in the database. Assume that $NFA(S, \delta)$ is very small with respect to 1, and that a shape S' is found at distance δ from S in the database. This match could not occur just by chance and is therefore a meaningful detection. Its explanation is usually the common origin of both shapes. Experimental evidence will show that $NFA(S, \delta)$ can be predicted accurately.

1. INTRODUCTION

Shape recognition is the field of computer vision which deals with the problem of finding out if a query shape lies or not in a shape database. This shape database is usually extracted from an image or a set of images. When we refer to “shape database”, we therefore refer to sets of images as well, along with a shape extraction algorithm.

The shape recognition task is usually split into three stages.

1. **Features extraction**, namely features detection [1, 2] then features grouping [3, 4].
2. **Pairing** of the request features [3, 5]: Pre-matching then accurate matching.
3. **Decision**. This step is certainly the key point of all shape recognition algorithms. Once two shapes are likely to match, how is it possible to come to a decision? As far as we know, the probabilities of matching, or the result of a voting process, are only used in order to classify the matches; the problem of absolute recognition thresholds has been raised by many authors [4, 6, 7].

Some authors have addressed the question of “wrong matches” occurring purely by chance (“estimating false alarms rates” [4], “conspiracy of random” [6], or “hallucinating a wrong fit” [7]). The proposed models did not lead, however, to an automatic recognition criterion, but only to an ordering criterion for matches.

Our plan is as follows. In section 2, we introduce the notion of *meaningful match*. In section 3 we briefly describe a shape feature extraction and encoding algorithm, based on the image level lines. A background model for the shape database and the decision step are explained in section 4. We test the correctness of the background model in section 5.

2. MEANINGFUL MATCHES

We shall first dress up an empirical statistical model of the shape database. The relevant matches will be detected *a contrario* as rare events for this model. This detection framework has been recently applied by Desolneux *et al.* to the detection of alignments [8] or contrasted edges [9], and by Almansa *et al.* to the detection of vanishing points [10]. It derives from a perceptual principle which states that if the expected number of appearances of some event by chance is very small, then it is highly *meaningful*. The main advantage of this technique is that the only parameter that controls the detection is the number of false alarms, a quantity that has a handy meaning.

Suppose that the problem is to decide whether a shape code matches some other shape code from a database of N_B codes. We assume that in our framework the shape codes are lists of n features, each of them belonging to a metric space (E_i, d_i) , $1 \leq i \leq n$.

Let $X = (x_1, \dots, x_n)$ be a query shape code, and let $Y = (y_1, \dots, y_n)$ denote an element of the cartesian product $E_1 \times \dots \times E_n$. Given a positive real number δ , we say that X and Y are δ -close if and only if:

$$\forall i \in \{1, \dots, n\}, d_i(x_i, y_i) \leq \delta.$$

Two codes match if they are δ -close, with δ *small enough*: we have to set a threshold for δ .

Assuming that all features are independent, we can compute the probability (denoted by $\mathcal{P}(X, \delta)$ in what follows) that a shape code Y is δ -close to X :

$$\Pr(Y \text{ s.t. } Y \text{ is } \delta\text{-close to } X) = \prod_{i=1}^n \Pr(y \in E_i \text{ s.t. } d_i(y, x_i) \leq \delta). \quad (1)$$

Each term of this product can be empirically estimated on a shape database. For each i , we compute the distribution function of $d_i(z, x_i)$, when z spans the i^{th} feature of the codes in the database. By this way, we can compute $\mathcal{P}(X, \delta)$.

Definition 1 (ε -meaningful match) $X = (x_1, \dots, x_n)$ being given, we say that a shape code $Y = (y_1, \dots, y_n)$ matches X ε -meaningfully if one has $N_B \cdot \mathcal{P}(X, d) \leq \varepsilon$, where $d = \max_{i=1, \dots, n} d_i(x_i, y_i)$.

Remark: Given $\varepsilon > 0$, one can compute the associated maximal distance $d^*(\varepsilon)$ such that $N_B \cdot \mathcal{P}(X, d) \leq \varepsilon$. This positive real number d^* is uniquely defined since the N functions $\delta \mapsto \Pr(y \in E_i \text{ s.t. } d_i(y, x_i) \leq \delta)$ (with $1 \leq i \leq n$) are non-decreasing, and so is their product; consequently, the function $\delta \mapsto \mathcal{P}(X, \delta)$ is pseudo-invertible. Each positive real d such that $d < d^*$ also satisfies $N_B \cdot \mathcal{P}(X, d) \leq \varepsilon$.

Notice that the empirical probabilities take into account the “rareness” or “commonness” of a possible match; indeed the threshold d is less restrictive in the first case and more strict in the other.

The following proposition shows that the number of false detections is controlled by ε . This provides a more intuitive way to control detections than just tuning the distance δ for each query.

Proposition 1 *The expectation of the number of ε -meaningful matches over the set of all shape codes in the database is less than ε .*

Proof: Let Y_j ($1 \leq j \leq N_B$) denote the possible shape codes, and χ_j denote the indicator function of the event e_j : “ Y_j matches ε -meaningfully X ”. Let $R = \sum_{j=1}^{N_B} \chi_j$ be the random variable representing the number of codes matching ε -meaningfully X . The expectation of R is $E(R) = \sum_{j=1}^{N_B} E(\chi_j)$. With definition 1, $E(R) = \sum_{j=1}^{N_B} \mathcal{P}(X, d)$, so $E(R) \leq \sum_{j=1}^{N_B} \varepsilon \cdot N_B^{-1}$, yielding $E(R) \leq \varepsilon$. ■

Remark: The key point is that we control the expectation of R . Since dependences between events e_j are unknown, we are not able to estimate the probability law of R . Nevertheless, the linearity still allows to compute the expectation.

Authors that have tried to deal with this problem [4, 7] have arbitrarily assumed that the events are independent.

To end with these definitions, we provide a measure for the quality of matching.

Definition 2 (Number of false alarms) *Given a shape code X and a distance $d > 0$, we call number of false alarms of a match with X at distance d the number*

$$NFA(X, d) = N_B \cdot \mathcal{P}(X, d).$$

The framework we presented provides a way to detect matches while controlling the number of objects that match purely by chance.

If we want that “random matches” between a query shape code and a database code appear only once on the average, we simply put $\varepsilon = 1$. If the query is made of N_Q equally relevant shape codes, and if we want to detect on the average at most one random match *over all query codes* (that is what we will do in section 5), we still put $\varepsilon = 1$ after replacing N_B by $N_B \cdot N_Q$ in definition 1 (in this case, proposition 1 obviously still holds).

Let us explain briefly the practical usefulness of formula (1): Our aim is to estimate the very small probability that a shape S' in a given database happens just by chance to look like a query S . This probability is intuitively very small if S is complicated enough. Thus, we cannot estimate empirically the probability of this event by just counting the shapes S' which are δ -close to S : there will simply be none. Thus, the trick is to decompose the computation of this very small probability into a product of observable probabilities. In order to do that, we shall in section 4 simply split the complicated shape S into a few simpler parts (in practice 5 parts). By doing so, we are able to observe for each single part, pieces of shapes in the database which are δ -close. In that way, we shall be in a position to learn from the database the probabilities on the right-hand side of (1).

3. SHAPE NORMALIZATION

Although this is not at all the aim of this communication, we have to briefly explain the coding procedure of shapes we used. This procedure provides structured information with as few arbitrary stages and thresholds as possible. See [3], for further details. Given an image:

1. Extract the maximal meaningful level lines (based on contrast [9]).
2. Smooth them by the Geometric Affine Scale Space [11]. We use a fast implementation [12].
3. Build invariant codes (up to either similitude or affine transforms). Following [3], we define local frames for each curve, based on robust directions (tangent line at inflexion points, or bitangent lines). A code is obtained by uniformly sampling a piece of curve in this frame.

4. BACKGROUND MODEL

We now explain how a code can be processed in order to fit the framework of section 2. Each code C is split into five pieces of equal length. This will lead us to define five independent variables associated with each code. This number five will be sufficiently large to attain very low values of the NFA. Each of these pieces is normalized by mapping the chord between its first and last points on the horizontal axis, the first point being at the origin: the resulting “small pieces of curve” are five features C_1, C_2, \dots, C_5 . For the sake of completeness, a sixth, global, feature C_6 is computed from the endpoints of the five previous pieces of the code.

The distances d_i between features, introduced in section 2, are the Hausdorff distance. Our *a contrario* model assumes that these six features are independent. Since the five pieces are separately normalized, this assumption is sound for random curves.

According to this model, if we observe that two codes match (in the sense of section 2), then a rare event has occurred. Now, in order to establish the link between the detected matches and shape recognition, we need to clearly identify and explain matches that are detected. There are two kinds of detections:

- *False Detections.* The detected matches are well described by the background model. These matches are realizations of the event: “the sets of independent features of the two codes are close enough”. *False Detections* are just detected because all their features are close “by chance”. According to definition 1, there should be, on the average, at most ε many of them.
- *Identifying (ID) and Non Identifying Detections (NID).* Any other detection should be an outlier of the background model. Therefore, it *must* be explained as a violation of the independency assumption. This violation is due to two factors:
 - Similar codes match because they were extracted from two similar (or identical) objects. We call detected matches of this type *Identifying Detections*.
 - Many shapes in images derive from natural or man-made objects having a common structure. For instance, many objects exhibit parallelism or width constancy, which yield strong deviations from the independency model. These *Non Identifying Detections* can be frequently detected (see figure 1).

Although being semantically different, these two types of detections cannot be distinguished from our point of view. Our method cannot control the number of NIDs since they are not *False Detections*, in the sense that if we take both shape codes out of their context, we would consider their match as correct.

5. EXPERIMENTS

As an experiment we check the detection thresholds on a very simple model: we take as database and query some random walks with independent increments. For size of databases in a wide range of values (from 100 to 10^6), the NFA is very well predicted. If we now make the same experiment on quantized level lines of an image of white noise, we find more than one false alarm, a few of which having a NFA close to 10^{-2} . This is a consequence of a violation of the independence assumption, due to the fact that level lines obey a global non-intersection constraint. In consequence, we shall only consider detections as valid if their NFA is below 10^{-2} , since detections with a NFA higher than 10^{-2} can be due to a slight violation of independence caused by the non-intersection constraint. Moreover, IDs always exhibit NFAs lower than 10^{-2} .

Figure 2 shows an example of shape matching with the presented method. The images can differ in such a way that a level line extracted from one of them could not be present or could be slightly different in the other one. That is why for example Saint George’s shoulder is not found.

Acknowledgements: This work was supported by the Office of Naval Research under grant N00014-97-1-0839, by the Centre National d’Études Spatiales, and by the Réseau National de Recherche en Télécommunications (projet ISII). We thank Jean-Michel Morel for valuable conversations and suggestions.

6. REFERENCES

- [1] C. Schmid and R. Mohr, “Local greyvalue invariants for image retrieval,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 5, pp. 530–535, 1997.
- [2] C. Carson, M. Thomas, S. Belongie, J.M. Hellerstein, and J. Malik, “Blobworld: A system for region-based image indexing and retrieval,” in *Proc. of the Int. Conf. on Visual Information Systems*, 1999.
- [3] J.-L. Lisani, L. Moisan, J.-M. Morel, and P. Monasse, “On the theory of planar shape,” *SIAM Multiscale Modeling and Simulation*, vol. 1, no. 1, pp. 1–24, 2003.
- [4] C. Olson and D.P. Huttenlocher, “Automatic target recognition by matching oriented edge pixels,” *IEEE Trans. on Image Processing*, vol. 6, no. 12, pp. 103–113, 1997.
- [5] Y. Lamdan and H. Wolfson, “Geometric hashing: A general and efficient model-based recognition scheme,” in *Proc. of the IEEE Int. Conf. on Computer Vision*, 1988.

- [6] W.E.L. Grimson and D.P. Huttenlocher, "On the verification of hypothesized matches in model-based recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 13, no. 12, pp. 1201–1213, 1991.
- [7] C.V. Stewart, "MINPRAN: a new robust estimator for computer vision," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, no. 10, pp. 925–938, 1995.
- [8] A. Desolneux, L. Moisan, and J.-M. Morel, "Meaningful alignments," *Int. J. of Computer Vision*, vol. 40, no. 1, pp. 7–23, 2000.
- [9] A. Desolneux, L. Moisan, and J.-M. Morel, "Edge detection by Helmholtz principle," *J. of Mathematical Imaging and Vision*, vol. 14, no. 3, pp. 271–284, 2001.
- [10] A. Almansa, A. Desolneux, and S. Vamech, "Vanishing point detection without any a priori information," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 4, pp. 502–507, 2003.
- [11] G. Sapiro and A. Tannenbaum, "Affine invariant scale-space," *Int. J. of Computer Vision*, vol. 11, no. 1, pp. 25–44, 1993.
- [12] L. Moisan, "Affine plane curve evolution: A fully consistent scheme," *IEEE Trans. on Image Processing*, vol. 7, no. 3, pp. 411–420, 1998.



Fig. 1. Non Identifying Detection on a book edge. 10^{-3} -meaningful match.



Fig. 2. Two different photographs from the same painting ("Saint George and the Dragon" by Uccello). Top: considered level lines. Left level lines (976 codes) are searched among right ones (38,669 codes). Bottom: 10^{-2} -meaningful matches. All are Identifying Detections (ID) as predicted by the theory.