Application of Deformable Template Matching to Symbol Recognition in Hand-written Architectural Drawings

Ernest Valveny, Enric Martí
Computer Vision Center - Computer Science Department
Universitat Autònoma de Barcelona
Edifici O - Campus UAB. 08193 Bellaterra. Spain
{ernest|enric}@cvc.uab.es

Abstract

We propose to use deformable template matching as a new approach to recognise characters and lineal symbols in hand-written line drawings, instead of traditional methods based on vectorization and feature extraction. Bayesian formulation of the deformable template matching allows combining fidelity to the ideal shape of the symbol with maximum flexibility to get the best fit to the input image. Lineal nature of symbols can be exploited to define a suitable representation of models and the set of deformations to be applied to them. Matching, however, is done over the original binary image to avoid losing relevant features during vectorization. We have applied this method to hand-written architectural drawings and experimental results demonstrate that symbols with high distortions from ideal shape can be accurately identified.

1. Introduction

Powerful and user-friendly CAD applications have been developed for helping in creating and modifying all kind of graphic documents. However, handwriting is still very widely used by designers as a more natural and easier technique to make preliminary drafts of them. It is therefore very desirable to develop interfaces capable of interpreting hand-written drawings and converting them to documents that can be modified with a CAD application [6].

Many approaches have been proposed for identifying and recognising symbols that can be found in these line drawings [1]. Most of them, however, work on printed graphics. In handwriting, due to its inherent imprecision, symbols can appear with very distorted and different shapes, and with additional, missing or noisy relevant features (lines, crossing and end points, etc.). In this context, methods based on skeletonization, vectorization and feature extraction may fail to identify correctly all symbols in the drawing.

Deformable template matching arises then as an alternative approach to represent, identify and recognise symbols in a hand-written graphic document, more flexible and better able to deal with all possible variations from ideal shapes.

In section 2 we justify the use of deformable template matching. In section 3 we give details about our representation for symbols and deformations, while section 4 explains the mathematical matching formulation. Section 5 shows some results obtained, and in section 6 we present conclusions from our work.

2. Deformable template matching as an alternative approach to symbol recognition

Existing approaches to identification and recognition of symbols in line drawings are generally based on the following steps [8]:

- Extraction of the primitives and features that can be found on the drawing, such as lines, curves, text boxes, solid regions, crossing and end points, etc.
- Recognition of symbols from grouping extracted primitives and features and comparing them with a model using some kind of pattern matching technique.

Methods based on this general approach decrease their efficiency and robustness as long as noise and distortion of symbols increase [8], as it is the case in hand-written drawings. Due to noise or distortion, some relevant primitives or features can be lost or erroneously detected. These errors are then propagated to the recognition step. Also, the great variety of shapes a symbol can take in handwriting, makes it very difficult to find a set or rules, or a general matching technique capable of identifying all of them. Figure 1 illustrates these drawbacks.

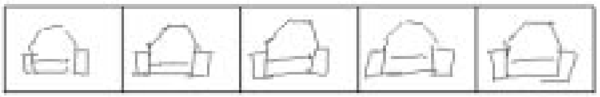

Figure 1. Vectorization of different instantiations of the same symbol.

As an alternative, in deformable template matching, symbol recognition starts from an image representing the ideal shape of the symbol. This initial image is modified according to a predefined set of deformation rules in order to adjust itself to the image to be recognised, but also keeping some degree of similarity to the ideal shape.

Deformable template matching has been applied to a great number of applications (see [2] for a review) where noisy, distorted or partial ocluded objects must be recognised, in many fields of computer vision. However, its application to document analysis is limited to the recognition of hand-written characters ([4][7][10]). We think it is also very suitable to recognise any kind of symbols in hand-written line drawings, basically because of these two main advantages:

- It works directly on the binary image and not on primitives and features extracted from it, avoiding so, problems arising by missdetecting some primitives.
- It has high capacity for reaching great distortions from an ideal shape without needing to model in advance all possible variations of a symbol.

3. Representation of symbols and deformations

Symbols that can be found in drawings are basically composed of lines. We can exploit this fact to define a specific model for symbols that consists in its representation by lines, which can be modified through geometric transformations (figure 2). This is an intuitive and natural way to represent symbols. Currently, we have restricted our model to straight lines. It must be noted that although symbol representation and deformations are expressed in terms of lines, matching is done over the binary input image.

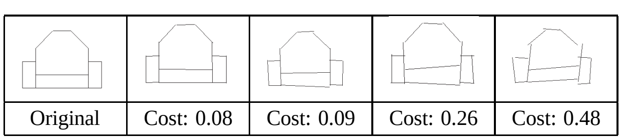

Figure 2. Deformations of a symbol.

We consider two kinds of possible deformations:

- Global deformations: this implies translation, rotation and scaling of the entire symbol. They do not have any associated cost because all of them must be equally probable.
- Local deformations of the lines that make up the symbol: each line can be translated, rotated and scaled separately from the other ones. This kind of deformations modifies the global shape of the symbol and allows adjusting it to the input image. To avoid changing too much the global shape of the symbol, each of these deformations must have an associate cost.

Figure 2 shows the lineal representation of a symbol and how it can be modified following the deformation model described before. Each deformation has its associated cost.

4. Matching between a symbol and an input image

Matching consists in finding the least possible deformation of a symbol that best fits to the input image. This can be modelled through the combined application of two opposite forces:

- One force that attracts the deformed model to the input image minimising the distance between the pixels of both images.
- Another force that prevents high deformations minimising a function that measures the degree of deformation.

The application of these two forces can be described using a Bayesian probabilistic framework [3], finding the deformation of the symbol that maximises the following expression:

$$P(D|I) = \frac{P(I|D) \cdot P(D)}{P(I)} \tag{1}$$

where D stands for any possible deformation of the symbol and I stands for the binary input image. $P(D|I)$ is the posterior probability that given an input image it really corresponds to a given deformation. $P(I)$ is the probability of observing a given input image. It does not affect the matching value. Next, we describe the other two terms involved in this formula.

4.1. Likelihood

$P(I|D)$ is the likelihood that given one deformation of the symbol, we can really observe the input image. It corresponds to the force that attracts the deformed model to the binary input image.

Taking advantage of the lineal nature of the model we define global distance between a deformed model and an input image as the weighted sum of distances between each line and the input image. Distance between a line an the image takes into account distance and difference of orientation between the line and the pixels at the contour of the image and, also, the degree of coincidence between the line and image pixels.

Then, likelihood can be defined in the following terms:

$$P(I|D) = \frac{1}{Z} e^{-Dist(I,D)} \tag{2}$$

$$\begin{aligned} Dist(I,D) \quad = \quad & \frac{1}{|L|} \sum_{l \in L} \Big(\frac{1}{n_l} \sum_{(x,y) \in l} (1 - e^{-d(x,y)} \cdot \\ & \cdot |\sin(\beta(x,y))|) + \alpha \frac{n_l - n_{Il}}{n_l} \Big) \end{aligned} \tag{3}$$

where exterior summation is over all lines in the symbol and interior summation is over all pixels in every line. $d(x,y)$ is the lowest distance between a pixel in the line and the contour of the image. $\beta(x,y)$ is the angle between the orientation of the line and the gradient angle of the nearest pixel in the contour. n_{Il} is the number of coincident pixels between the line and the input image. n_l is the number of pixels of the line. α is a weighting factor.

4.2. Prior probability

$P(D)$ represents the prior probabiliy distribution of deformations, the probability that a given deformation of a symbol is still a valid representation of that symbol. It can be viewed also as the application of the force that prevents high deformations from the ideal shape.

In the expression for this prior probability we assume:

- Each possible deformation (translation, rotation or scaling) of a particular line of the model follows a gaussian distribution of mean zero, every one of them with different standard deviation.
- All different deformations of all lines of the model are independent of each other.

Under these assumptions we get this expression for prior probability:

$$P(D) = \prod_{l \in L} \prod_{i \in T} \frac{1}{2\pi\sigma_{il}^2} e^{-\frac{\Delta_{il}^2}{2\sigma_{il}^2}} \tag{4}$$

where l represents any line of the symbol, i represents each possible tranformation (rotation, translation, scaling) of a line, σ_{il} is the standard deviation of transformation i for line l, and Δ_{il} is the amount of transformation i for line l.

4.3. Cost function

Developing expressions (1), (2), (3) and (4) as explained in [3], we get the following goal function $F(I,D)$ to be minimised:

$$F(I,D) = Dist(I,D) + \sum_{l \in L} \sum_{i \in T} \frac{\Delta_{il}^2}{2\sigma_{il}^2} \tag{5}$$

where the first term models the attraction between the image and the model, and the second term models the attraction between the deformed model and its original shape. Lower values of σ_{il} will result in more rigid templates, while higher values of will allow more distorted symbols to be identified, but some symbols might be confused.

This function can be minimised using a simulated annealing algorithm [5]. Starting from the ideal model of the symbol, each step of the algorithm randomly generates a new deformation of every line and accepts it depending on the difference in the function cost, until convergence is reached. As a result, we get the set of deformations (translations, rotations and scaling) for each line, which makes the model adjust better to the input image. The complexity of the algorithm is high, but allows to find a solution close to global minima.

5. Experimental results

We have applied our deformable template matching model to the recognition of symbols in hand-written architectural drawings.

Figure 3 illustrates the recognition of some hand-written symbols with different types of distortions, noise, and changes of position and orientation in lines. We can see how the initial model of the symbol is adjusted to the input images (superimposed on gray over the input image). For each symbol the final minimum cost found by the matching algorithm is presented. The graphic demonstrates that the final cost discriminates different symbols from each other.

Figure 4 shows the result of applying the matching of the model of a symbol to images of other symbols. It can be seen how the algorithm cannot find a deformation that fits images different from the initial symbol, and how the final cost for those images is very much higher.

Figure 5 shows the application of matching to dimension symbols recognition. Dimensions play an important role in hand-written drawings [9], but due to its small size, distortions or noise make it difficult to recognise them

Finally, figure 6 is an example of how this method could be applied to the identification of symbols in small areas of an entire drawing, allowing thus to develop a technique to locate and recognise all symbols in the drawing.

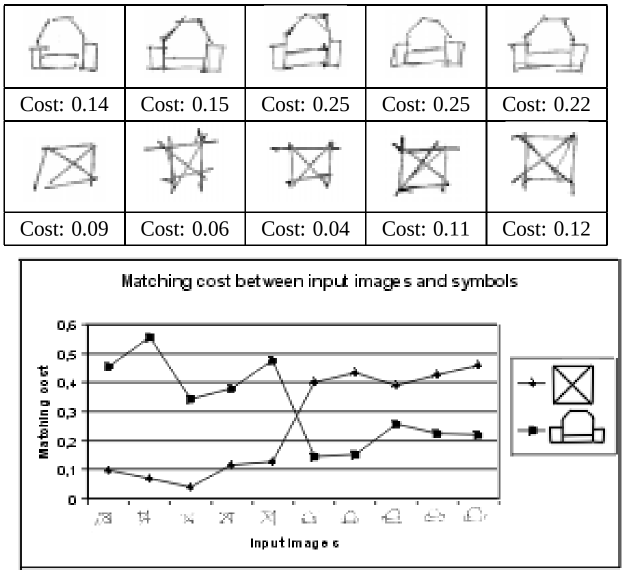

Figure 3. Recognition of different images of symbols.

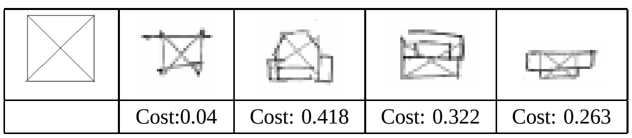

Figure 4. Matching of symbols with input images corresponding to different symbols.

6. Conclusions and future work

We have developed a deformable template matching model to recognise lineal symbols. Unlike methods based on previous vectorization of images, our model works directly on binary images avoiding loss of information due to noise and distortion. So, it is more flexible and capable of handling deformations caused by handwriting.

We have taken advantage of the lineal nature of symbols representing them by straight lines. Deformations very close to those produced by handwriting are generated by simply applying geometric transformations over the lines.

Bayesian formulation of the problem allows developing the matching procedure as a minimisation of a cost function composed of two terms, deformation cost and distance cost, which keeps balance between fidelity to original shape and proximity to input image. We have used simulated annealing to minimise this function.

Results from symbol recognition in hand-written architectural drawings show that the model can represent very well symbols produced by handwriting and that high deformations can be identified.

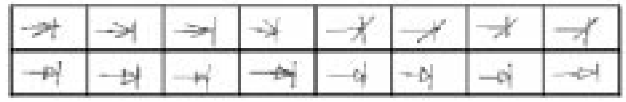

Figure 5. Identification of dimension symbols.

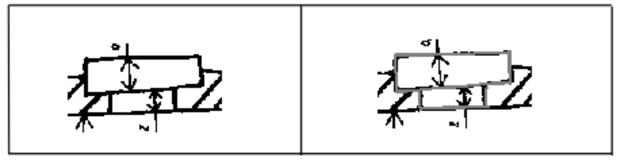

Figure 6. Searching a symbol in a small area of a drawing.

We are working now on the generalisation of the method to other types of primitives, and on the simplification of the cost function to make it easier to minimise.

References

[1] A. Chhabra. Graphic symbol recognition: An overview. In K. Tombre and A. Chhabra, editors, *Graphics Recognition: Algorithms and Systems*, pages 68–79. Springer Verlag, Berlin, 1998.

[2] A. Jain, Y. Zhong, and M. Dubuisson-Jolly. Deformable template models: A review. *Signal Processing*, 71:109–129, 1998.

[3] A. Jain, Y. Zhong, and S. Lakshmanan. Object matching using deformable templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(3):267–277, March 1996.

[4] A. Jain and D. Zongker. Representation and recognition of handwritten digits using deformable templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(12):1386–1391, December 1997.

[5] P. V. Laarhoven and E. Aarts. *Simulated Annealing: Theory and Applications*. Kluwer Academic Publishers, 1989.

[6] J. Lladós, J. López-Krahe, and E. Martí. Hand-drawn document understanding using the straight line hough transform and graph matching. In *Proceedings of the 13th IAPR International Conference on Pattern Recognition*, volume 2, pages 497–501, Viena, August 1996.

[7] M. Revow, C. Williams, and G. Hinton. Using generative models for handwritten digit recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(6):592–606, June 1996.

[8] K. Tombre. Graphics recognition - general context and challenges. *Pattern Recognition Letters*, 16(9):883–891, 1995.

[9] E. Valveny. Implementació d'un mòdul d'anàlisi de cotes en dibuixos a mà alada dins d'un Àmbit d'arquitectura. Master's thesis, Universitat Autnoma de Barcelona, Barcelona, 1994.

[10] T. Wakahara. Shape matching using lat and its application to handwritten numeral recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6):618–629, June 1994.