Symbol spotting for technical documents : An efficient template-Matching approach

Jonathan Weber and Salvatore Tabbone
Université de Lorraine, LORIA-UMR 7503, Vandœuvre-lès-Nancy, F-54506, France
{jonathan.weber, tabbone}@loria.fr

Abstract

Symbol retrieval for technical documents is still a hot challenge in the document analysis community. In this paper we propose another way to spot symbols. A pixel-based template operator which is an adaptation of the hit-or-miss transform is defined. This operator is robust to translation, rotation and reflection. Experimental results on a real application show the efficiency of our approach.

1 Introduction

Symbol spotting is an emerging topic and few works haven been proposed so far. Although well-known symbol descriptors work well to describe isolated symbols, their performance in real applications drop away when symbols are embedded in documents. In a spotting system the user submits a query he wants to retrieve from the document databases and the system retrieves zones into the documents likely to contain the query. The query is defined by the user, usually it is a cropped image on a document belonging to the database or hand-sketched. Broadly, symbol spotting methods are defined into pixel and structural decomposition. A common scheme is to decompose the document into components and then apply a descriptor on each component. The decomposition could be focused locally on points or regions of interest or globally through a structural representation. Usual representations are key-points or lines which are defined on segmentation steps and the quality of the spotting methods are strongly dependent on the quality of the prior segmentation. For structural methods a vectorization step is needed and for these reasons, usually works are focused on synthetic documents since it is well known that vectorization methods are not robust to noise and cluttered background. In this perspective, polyline primitives [10] have been proposed to recover the problem of segments fragmentation due the sensitivity to noise of raster to vector algorithms. Graph representation [2, 12, 8] remains the most popular structural structure since it offers a more powerful representation to encode the relationship (parallelism, adjacency, straight angle...) between the different subparts of a symbol but the final results are still strongly dependent on the information we put into the graph. Keypoints approaches usually based on corner points [6] are more robust to noise but they describe locally the document and there is a need to organize these points spatially to validate hypothesis to find a symbol [9].

In our opinion, methods based on correlation principles [3, 1] are more likely to locate accurately positions of a queried symbol and are especially robust to be applied on real applications. In these methods documents are decomposed into regions following a grid partition or a sliding window and the similarity (like a correlation function) with the query is measured as maxima on a surface of normalized correlation. However, the invariance to symbol variations (size and rotation) remains a bottleneck.

In this paper we propose a template matching operator HMTAIO (Hit-or-Miss Transform Adapted to Information Overlapping) which is an adaptation of the hit-or-miss transform. The advantage of our approach compared to correlation methods is its robustness to occlusion and overlapping. Moreover, our method is enough robust to be applied on real documents unlike the majority of previous ones. Contrary to synthetic documents, in real documents the segmentation is difficult, the information is dense, cluttered and often occluded.

2 Recall of the Hit-or-miss transform

The Hit-or-Miss Transform (HMT) has been first defined for binary images. In such images, this operator [11] is quite trivial and uses two disjoint structuring elements: the first has to match the foreground while

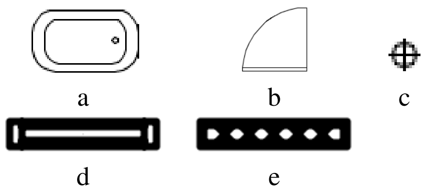

Figure 1. Some query symbols (for visualization purpose grey-levels are inverted).

the second has to match the background. Both matches are necessary to give a positive matching response. Extension of the HMT operator to grey-level images led to several definitions which have been reviewed and unified by Naegel *et al.* [5]. These extensions of binary hit-or-miss also need a perfect matching of their structuring elements or structuring functions which is problematic when dealing with noisy images or information overlapping. So, new extensions able to deal with noisy images have been proposed [7, 4]. Their robustness to noise is achieved with a fuzzy fitting. Basically, the structuring elements do not have to match the whole background and foreground but only a percentage of them. This percentage is set by a parameter and such approaches have proven their efficiency on very noisy images.

But, these fuzzy HMT are not adapted to our problem, in fact in technical document we are more facing information overlapping than noise. In this purpose, we cannot use a global matching percentage of background/foreground pixels. We have to treat differently foreground and background parts of our template. So we propose a new approach in the next section.

3 HMTAIO operator for symbol spotting

In this section, we recall the specific need of the symbol spotting in technical plan, introduce our new HMT operator adapted to information embedded into the document. Then we explain how the two structuring functions are automatically designed and present some implementation details.

3.1 Motivation

The major symbol spotting difficulty when dealing with plan is the overlapping/occlusion of symbols which are embedded into the document and make harder their spotting. In figure 2, we can observe that two symbols are connected which keep us from using primitives analysis for the spotting. Indeed, the two symbols are overlapped, this makes difficult a spotting

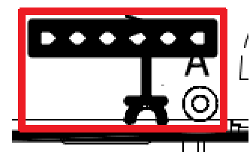

Figure 2. Case of symbol overlapping.

method based on structural or keypoints decomposition. Then, we choose to use a pixel-based template-matching operator like HMT. The operator has to be robust to translation, rotation and reflection. In fact, a symbol can have various positions and orientations. Moreover, it can be a reflection of the queried symbol. Often, the operator has not to be robust to scale change.

The HMT meets these requirements, it is naturally robust to translation and robustness to rotation and reflection can be achieved by geometric transformations of the template represented by the structuring functions.

The information overlapping in plan implies that pixel values in the document can be higher than the query symbol pixel values (especially for the background) as other symbols or information can be in the vicinity of the symbol.

3.2 Definition

To achieve the robustness to information overlapping, we define a new operator HMTAIO (HMT Adapted to Information Overlapping) which checks that the grey-level in the image is similar to the grey-level of the foreground structuring functions (F) and checks that the grey-level in the image is similar to the grey-level of the background structuring functions (B). The adaptation to information overlapping consists in heavily impacting the matching score if the foreground is not fitted (high α) and slightly impacting the score if the background is not totally fitted due to information overlapping (low β). This leads to the following definition of HMTAIO :

$$HMTAIO_{F,B}(I)(x) = \frac{HMTAIO_F(I)(x)^{\alpha}}{2} + \frac{HMTAIO_B(I)(x)^{\beta}}{2} \quad (1)$$

where

$$HMTAIO_F(I)(x) = \frac{\sum_{p \in F} \min\{1, \frac{I(x+p)}{F(p)}\}}{card(F)} \quad (2)$$

and

$$HMTAIO_B(I)(x) = \frac{\sum_{p\in B}\min\{1, 1 - \frac{I(x+p)-B(p)}{I^{\max}-B(p)}\}}{card(B)} \tag{3}$$

where I is an image and $I^{\max}$ is the highest grey-level of the image I. α and β are empirically set to 3 and 1.

The matching score for the foreground is the sum of individual matching score of each foreground pixel divided by the number of foreground pixel. For each pixel, the matching score is 1 if the image pixel value is equal or greater than the corresponding foreground pixel value otherwise the matching score is in $[0; 1[$ allowing a robustness to small value differences. The idea is the same for background pixel but there, the image pixel value has to be lower or equal to the corresponding background pixel value in order to be perfectly fitted with a matching score of 1, otherwise the matching is also $[0; 1[$.

We obtain matching score for foreground and background between 0 and 1 which we combine to obtain a global matching score. As said, we set $\alpha = 3$ in order to heavily impact the global score in case of bad matching of the foreground. In fact, as we noticed previously, the overlapping does not induce smaller pixel values but only higher pixel values, if other information has higher pixel value. So, if the foreground matching score is not 1, there is some missing foreground information and we obtain a low global matching score. On the contrary, due to the overlapping, the background can be badly matched if there are other symbols or information near the symbol to spot. We have to be robust to such situations and we set $\beta = 1$ for the background matching score. We will show that experimental results are promising using these values.

The result is a matching score map. To obtain the spotted symbols, the user has to interactively set a threshold on the matching score. It is a very intuitive and quick step.

3.3 Automatic structuring functions design

The design of structuring functions for a HMT operator is often a complex task. It requires some knowledge from the user. Here, we propose an automatic structuring functions design where the user selects a part of the plan containing the symbol (cf. figure 3.a).

From this query, we have to build two templates, one for the foreground (see figure 3.b) and the other for the background (see figure 3.c), called structuring functions which are basically spatial and spectral definitions

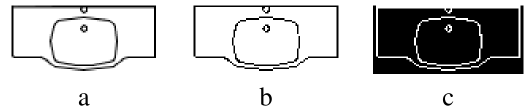

Figure 3. Structuring functions automatically built from the query image : (a) query image, (b) foreground, (c) background.

of the symbol to match. The design of the structuring functions consists in labeling each pixel as foreground or background pixels. The label of each pixel of the query is obtained as follows :

$$\text{label}(I)(R)(p) = \begin{cases} F & \text{if } R(p) > \frac{I^{\max}}{2} \\ B & \text{otherwise} \end{cases} \tag{4}$$

where R is the query image.

Thus, the structuring functions are created without any user interaction.

3.4 Implementation details

In order to be robust to rotation and reflection, we apply the HMTAIO using original structuring functions and their reflection with different orientations. The final HMTAIO score is the highest obtained with the different variations of the structuring functions.

Computing HMTAIO on large plan can be computationally expensive. A great advantage of HMTAIO is that matching score obtained for one pixel of the plan is independant from the matching score obtained on other pixels, so it can be done in parallel. While current processors are mostly multi-core, it allows to heavily reduce the computation time of the spotting process.

4 Experimental results

We perform experiments on a real plan of 13,979 × 9,871 (137,986,709 pixels). We test our symbol spotting approach on 5 different queries (see Fig. 1).

The results presented in Table 1 are obtained by using the smallest matching score threshold which does not produce false positive. We observe that our new operator has a really good spotting rate (98-100%) even with very different types of symbol (e.g. thin, bold, small, large, ...). The computation times are obtained with a Java implementation on a Core i7 Q720 laptop CPU using 8 parallel threads. The important computation time difference between the queries is due to their different sizes. The computation time is important but it can be heavily reduced by using multi-processors architecture. Moreover, this computation time can be achieved offline. The online part which is the selection of the best threshold by the user is achieved in real-time.

Table 1. Spotting results obtained with 5 different queries.

Query	Thresh.	TP	FN	Time (sec)	Spotting rate
a	0.57	80	0	12, 340	**100%**
b	0.71	141	2	35, 750	**99%**
c	0.77	91	2	699	**98%**
d	0.91	24	0	15, 319	**100%**
e	0.89	227	2	15, 060	**99%**

The figure 4 illustrates the spotting efficiency of our approach in complex situations. The first example shows that our approach is able to spot a symbol even if it is connected to another symbol. The second example highlights the ability of HMTAIO to extract a symbol even if it is highly overlapped by other information. The last one underlines the discrimination power of our approach which spots the queried symbol and not the other symbol even if it is very similar to the query.

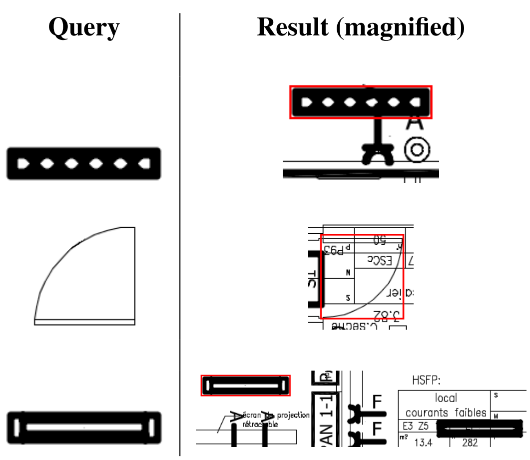

Figure 4. Difficult spotting cases.

5 Conclusion and perspectives

In this article, we have proposed a new template-matching operator dedicated to the symbol spotting in technical plan, the HMT Adapted to Information Overlapping (HMTAIO). We have shown its efficiency, even in dense and cluttered information cases. We also propose a method for the automatic design of structuring functions.

Our future works will focus on the automatic setting of the parameters of our approach in order to obtain a fully automatic symbol spotting method. Moreover, we have to plan more experiments (other symbols, other technical documents) to fully validate our method.

References

[1] S. Escalera, A. Fornés, O. Pujol, J. Lladós, and P. Radeva. Circular blurred shape model for multiclass symbol recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 41(2):497–506, April 2011.

[2] J. Fonseca, A. Ferreira, and J. Joaquim. Content-based retrieval of technical drawings. *International Journal of Computer Applications in Technology*, 23(2-3):86–100, March 2005.

[3] W. J. MacLean and J. K. Tsotsos. Fast pattern recognition using normalized grey-scale correlation in a pyramid image representation. *Machine Vision and Applications*, 19(3):163–179, 2008.

[4] P. Murray and S. Marshall. A new design tool for feature extraction in noisy images based on grayscale hit-or-miss transforms. *IEEE Transactions on Image Processing*, 20(7):1938–1948, 2011.

[5] B. Naegel, N. Passat, and C. Ronse. Grey-level hit-or-miss transforms - part 1: Unified theory. *Pattern Recognition*, 40(2):635–647, February 2007.

[6] T.-O. Nguyen, S. Tabbone, and A. Boucher. A symbol spotting approach based on vector model and a visual vocabulary. In *10th International Conference on Document Analysis*, pages 708–712, 2009.

[7] B. Perret, S. Lefèvre, and C. Collet. A robust hit-or-miss transform for template matching applied to very noisy astronomical images. *Pattern Recognition*, 42(11):2470–2480, 2009.

[8] M. Rusiñol, A. Borrás, and J. Lladós. Relational indexing of vectorial primitives for symbol spotting in line-drawing images. *Pattern Recognition Letters*, 31(3):188–201, 2010.

[9] M. Rusiñol and J. Lladós. Word and symbol spotting using spatial organization of local descriptors. In *8th IAPR Workshop on Document Analysis Systems*, pages 489–496, 2008.

[10] M. Rusiñol, J. Lladós, and G. Sánchez. Symbol spotting in vectorized technical drawings through a lookup table of region strings. *Pattern Analysis and Applications*, 13(3):321–331, 2010.

[11] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, 1982.

[12] L. Wenyin, W. Zhang, and L. Yang. An interactive example-driven approach to graphics recognition in engineering drawings. 9(1):13–29, 2007.