Main problems and proposed solutions for improving Template Matching

Khalid Aznag*, Ahmed El Oirrak*, Essaid El Bachari*

* Computer Science Department, Cadi Ayyad University, Marrakesh, Morocco
E-mail: khalid.aznag@ced.uca.ac.ma

Abstract—In this work we discuss the problems of template matching and we propose some solutions. Those problems are: 1) Template and image of search differ by a scale, 2) Template or image of search is object of rotation, 3) Template or image of search is object of an affinity. The well known method is NCC (Normalized Cross Correlation); this method can not handle scale, rotation, affinity or occlusion. Also the NCC is not preferred for binary image. So we propose here to use index similarity for example Jaccard index.

Keywords—Binary image, Template matching, Similarity index, NCC.

I. INTRODUCTION

With the advent of the first computers develop the first systems OCR (Optical Character Recognition) in years 60 (These include the Research Center Thomas J. Watson IBM New-York [1], or the work of Bell Telephone Labor [2], These systems operate essentially on typeset characters and using techniques known template matching, template matching is a method of research and finds the location of an image (template) in a source image. Template Matching techniques compare images pixel by pixel using statistical classifiers. Also template matching is one of the most common techniques used in signal and image processing [3], and used in computer vision, in image processing such as image retrieval [4], image recognition [5], image registration [6], object detection [7] and stereo matching [8]. Normalized Cross-Correlation (NCC) [9] [10] and Zero Normalized Cross Correlation (ZNCC) [11] similarity functions are widely used in template matching as well as in motion analysis, stereovision, etc. Despite advantages of template matching we found problems in this technique for instance scaling, rotation and affinity. In this work we propose some solutions to solve these problems and we propose a new approach based similarity index.

The rest of the paper is organized as follows:

We recall in the second section NCC method. In section 3 we present our approach based on the similarity index. In section 4 we describe the new proposals approaches, Section 5 contains the conclusion. Finally Future works in section 6

II. NCC METHOD

Cross-correlation is the comparison of two different time series to detect if there is a correlation between metrics with the same minimum and maximum values. For example: “Are two audio signals in phase?”. In fact, Normalized cross-correlation is also the comparison of two time series, but using a different scoring result. Instead of simple cross-correlation, it can compare metrics with different value ranges [12].

A. Image of the same support

If the two images have exactly the same support I, \( n_{ij} \) is defined as \( f(x, y) = i \) and \( f_0(x, y) = j \) with \( i, j \in \{0,1\} \) and \( f_0 \) describing the model of a form and \( f \) application describing an unknown form whose characteristics are searched for, it follows the equation:

\[
\sum_{(x,y)\in I} |f(x,y) - f_0(x,y)| = n_{01} + n_{10} \tag{1}
\]

To compare two images pixel by pixel the distance used is the Hamming distance or Manhattan.

\[
D_{Manhattan}(F, F_0) = \sum_{(x,y)\in I} |f(x,y) - f_0(x,y)| = n_{01} + n_{10} \tag{2}
\]

The disadvantage of this approach is its extreme sensitivity to noise and any affine transformation of the form.

B. Image of different support

If the supports are different, but they contain as the shapes have the same size, for example \( I_0(n_{01},m_0) \) is smaller than
\( I(n,m) \) \((n_0 < n \text{ and } m_0 < m) \) then a matrix correlation matrix defined on \((n-n_0,m-m_0)\) can be used:
\[
M[i,j] = \frac{\sum_{(k,l) \in \mathbb{U}} f_0(k,l) f(k+i,l+j)}{\sum_{(k,l) \in \mathbb{U}} (f(k+i,l+j))^2}
\] (3)

The values of \( M \) highest indicate the positions of \( I \) where \( f \) is similar to \( f_0 \). The description obtained is not Vector but Matrix; it is essentially used to find a pattern in a scene. In this method can not compare models with different sizes or rotated.

### III. Binary Image and Similarity Index

Jaccard index (JI), also known as the Jaccard similarity coefficient [13], is a measure used to identify the degree of similarity and diversity of two data windows. Consider two data windows \( X_i \) and \( Y_j \). The measurements of degree of coefficient overlap between two windows is done by calculating the ratio of the number of attributes shared \( X_i \) and \( Y_j \). For simplicity, consider two sets \( A \) and \( B \) in place of data windows. The intersection \( A \cap B \) and union \( A \cup B \) between these two sets can be measured according to set theory. Thus, the Jaccard index is calculated as follows:
\[
\text{Jaccard}(A,B) = \frac{(A \cap B)}{(A \cup B)}
\] (4)

To determine the similarity between two data windows \( X_i \) and \( Y_j \), Intersection between these windows is obtained by computing the cardinality of the attribute value in the window as shown in the equation 5:
\[
\tilde{X}_i \cap \tilde{X}_j = \sum_{k=1}^{n} X_{ij} \cap X_{kj}, X_{ij} \in \tilde{X}_i \text{ and } X_{kj} \in \tilde{X}_j
\] (5)

Where,
\[
X_{ij} \cap X_{kj} = \begin{cases} 1, & \text{if } X_{ij} = X_{kj} \\ 0, & \text{else if} \end{cases}
\] (6)

Theoretically standard sets, the union of two sets is the set of all distinct elements in sets. Although the index calculation Jaccard similarity measure The Union between two data windows \( X_i \) and \( Y_j \) is obtained using the equation 7:
\[
\tilde{X}_i \cup \tilde{X}_j = \sum_{k=1}^{n} X_{ij} \cup X_{kj}, X_{ij} \in \tilde{X}_i \text{ and } X_{kj} \in \tilde{X}_j
\] (7)

Thus, measuring Jaccard similarity or Jaccard index, \( Jaccard(\tilde{X}_i,\tilde{X}_j) \) between two the windows \( \tilde{X}_i \) and \( \tilde{X}_j \) is:
\[
Jaccard(\tilde{X}_i,\tilde{X}_j) = \frac{\text{cardinality of Intersection}}{\text{cardinality of Union}}
\] (8)

### IV. Experimental Results

In this section we apply NCC and similarity method using the following steps:

- Step 1: Loading template and the search image. Charged images are color images. We converted these images into binary images,
- Step 2: We calculate the index of similarity or NCC between two blocks on the template and search image,
- Step 3: From the value calculated in step 2, the image location is determined (see Figures 1.c).

#### Table 1

<table>
<thead>
<tr>
<th>Images</th>
<th>Similarity index</th>
<th>Value of the correlation function (NCC method)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1.a,1.b)</td>
<td>0.8896</td>
<td>0.9150</td>
</tr>
</tbody>
</table>

In Table 1 we see that our approach based on the similarity index is also effective for colored images. We find that our approach based on the similarity index is applicable for binary and color images. Regarding the case of the image correlation, when the two images are identical correlation function tend to 1, where the images are different the function value approaches tend to 0.

### V. Proposition of Other New Techniques to Solve Scale, Rotation, Affinity Problems

#### A. Scale Problem

Once we find the regions in each image we extracted invariant relative to the scale change (see Figure 2), then we compare invariants using Hausdorff distance [14], Frechet distance [15], etc because the invariants are not the same size.
We extract objects from $I (R_1, R_2, \ldots)$. And we compute Transform Fourier $[16] \quad TF(R_1), TF(R_2), \ldots$ also we compute $TF(T)$. We take $\frac{|TF(R_i)|}{\max TF(R_i)}$ as invariant. So we compare $\frac{|TF(T)|}{\max TF(T)}$ with $\frac{|TF(R_i)|}{\max TF(R_i)}$.

During this translation problem, we can see a translation of the letter "B" in template image, in the same way it detects objects (Figure 3.c), then we calculate distance with Fourier invariants.

Let $T$ denote template image and $I$ denote search image. We extract objects from $I (R_1, R_2, \ldots)$. And we compute Transform Fourier $[16] \quad TF(R_1), TF(R_2), \ldots$ also we compute $TF(T)$. We take $\frac{|TF(R_i)|}{\max TF(R_i)}$ as invariant. So we compare $\frac{|TF(T)|}{\max TF(T)}$ with $\frac{|TF(R_i)|}{\max TF(R_i)}$.

During this rotation problem, we can see a rotation of the letter "B" in template image, in the same way it detects objects (Figure 3.c), then we calculate distance with Fourier invariants.

During this affinity problem, we can see an affinity between the objects in the template and search images (Figure 3.c), then we calculate distance with Fourier invariants.

We extract objects from $I (R_1, R_2, \ldots)$. And we compute Transform Fourier $[16] \quad TF(R_1), TF(R_2), \ldots$ also we compute $TF(T)$. We take $\frac{|TF(R_i)|}{\max TF(R_i)}$ as invariant. So we compare $\frac{|TF(T)|}{\max TF(T)}$ with $\frac{|TF(R_i)|}{\max TF(R_i)}$.

During this affinity problem, we can see an affinity between the objects in the template and search images (Figure 3.c), then we calculate distance with Fourier invariants.

We extract objects from $I (R_1, R_2, \ldots)$. And we compute Transform Fourier $[16] \quad TF(R_1), TF(R_2), \ldots$ also we compute $TF(T)$. We take $\frac{|TF(R_i)|}{\max TF(R_i)}$ as invariant. So we compare $\frac{|TF(T)|}{\max TF(T)}$ with $\frac{|TF(R_i)|}{\max TF(R_i)}$.


