Using Normalized Cross-Correlation for Object Recognition

Vladimir Nikolin 320508369
Supervisor: Samuel Kosolapov
August 9th, 2015

In fulfillment of the excellence program
ORT Braude College, Karmiel

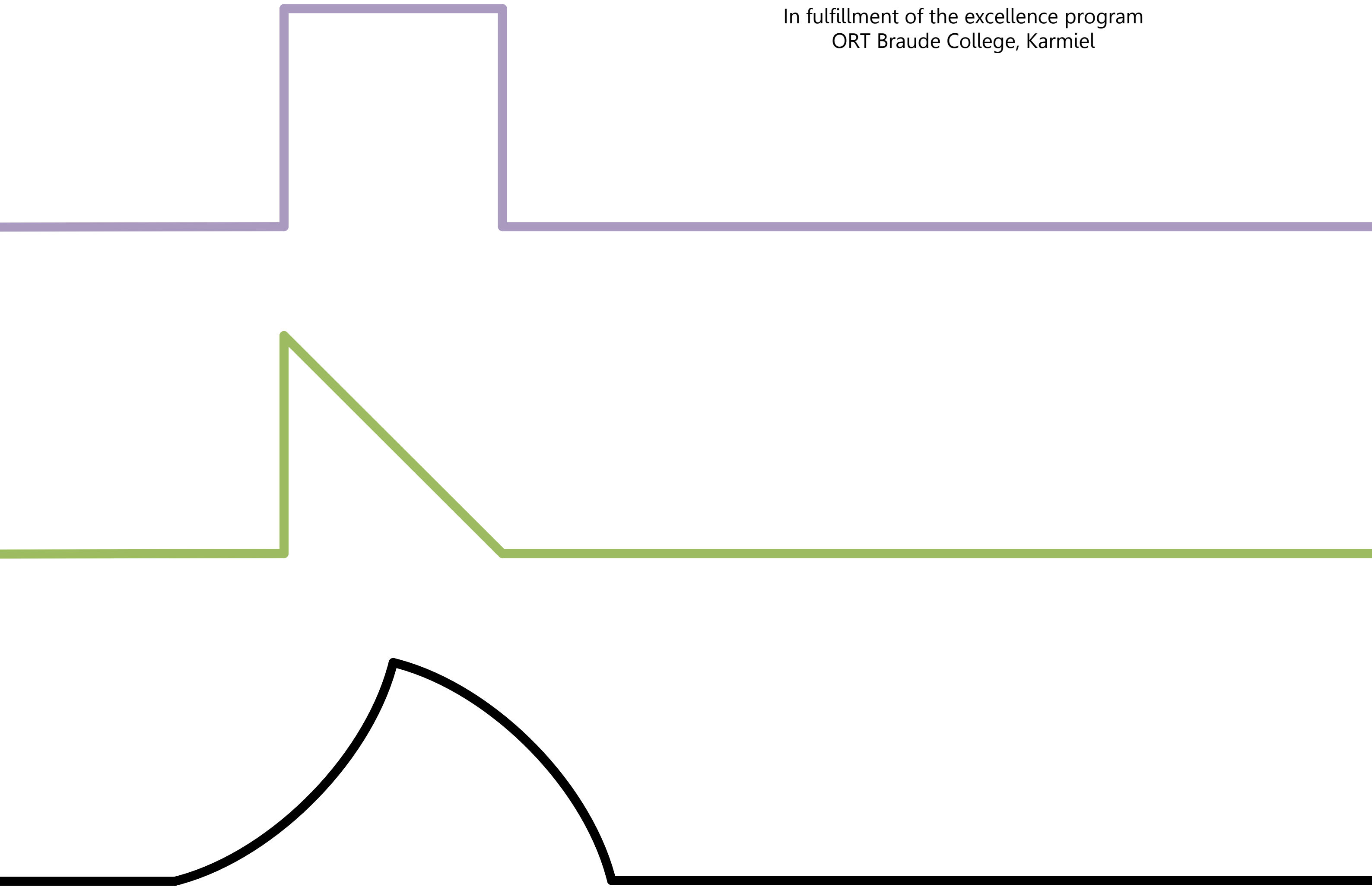


TABLE OF CONTENTS

1. INTRODUCTION 3
2. BACKGROUND 3
2.1. Cross-Correlation 3
2.2. Normalized Cross-Correlation 6
3. NCC OBSERVATIONS 7
3.1. Template Matching Example 8
3.2. Normalization Advantages 9
3.3. Multiple patterns presence 10
3.4. Scale Variance 10
3.5. Rotational Variance 11
4. CONCLUSION 12
5. Appendix 1 13
6. Appendix 2 14
7. REFERENCES 15
8. FIGURES INDEX 16

1. INTRODUCTION

The mathematical correlation function is used today in vast array of domains to measure similarity over various data, continuous or discrete. The prominent use is in signal processing applications. Correlation applicable to digital imaging, given its discrete, pixel-wise nature. In this work, we want to introduce and explain the way Cross-Correlation (CC) between two signals is calculated over a given lag and how it is normalized to produce more robust results. We are interested in computer vision and template matching capabilities of the Normalized Cross-Correlation (NCC). For that we will see how to expand NCC for use in two dimensional computer imaging space.

Keywords: Computer vision, Correlation, Feature matching

2. BACKGROUND

The human brain has a natural ability to recognize familiar features in our eyesight. Images captured by the eye are matched, almost instantaneously, to the information in the brain. Most of the modern image recognition techniques attempt to imitate the human eye recognition in a digital manner, to spot distinct features in an image. Some of the more popular methods used are SAD (Sum of Absolute Differences), SSD (Sum of Square Differences), CC (Cross-Correlation). In this paper, we will introduce and examine the CC method and its application in template matching. Moreover, we will examine the use of CC method in digital imaging domain.

The basic idea of template matching is pixel-wise operations and comparison over digital images. Digital image, being essentially a two-dimensional array of pixels, which colors are represented as discrete, integer values. At the most basic level, if we search for a certain pixel value range in an image, we will register a certain color in that image.

We want to detect features. A feature can be described as a group of pixels or a sub-frame, containing pixels of certain values and arranged in a certain order (not necessarily an order feasible for the human eye). We would like to compare such pixel arrangement against a reference, usually bigger, image in order to spot the given feature. Or, to conclude that the feature is not present. That is, if went through the whole reference frame comparing the group of pixels without a match.

2.1. Cross-Correlation

By definition, Cross-Correlation (CC) is a mathematical measure of similarity between two signals (Signals as continuous data, or arrays as discrete data), at a different lag positions (times), thus it is also known as “sliding inner product”. In a similar fashion, CC is used to spot a specific sub-pattern in a broader data space: Finding a known signal in a noisy one, or spotting cyclic data. CC is one of the most popular sub-pattern registration methods. CC is vastly used in Computer Vision, Cryptanalysis, Particle Analysis and more.

CC is applicable in a continuous data domain (i.e. Signal Processing) where it’s calculated as:

$$\mathrm{CC}(a,b,d) = \int_{-\infty}^{\infty} a^*(t)g(t+d)dt = (a \star b)(d) \tag{1}$$

Where a is the source signal, b is the searched sub-signal d is the time lag and a^* denotes the complex conjugate of the function a.

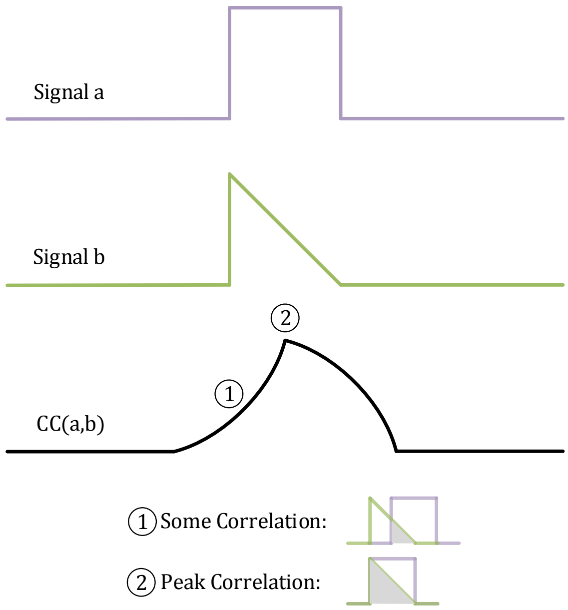


Fig. 1 Visualization of Cross-Correlation between two signals

The cross-correlation is similar in nature to the convolution of two functions, which has the delay factor inverted.

As we are interested in working with discrete natured pixels in a digital image, we will focus on a discrete data domain implementation on CC: For two discrete sets a, b, Cross-Correlation coefficient at a delay (time lag) d is defined [1] as:

$$\mathrm{CC}(a,b,d) = \sum_i a(i)b(i-d) \tag{2}$$

Let's examine the equation. The sum runs the two arrays with different lag at each iteration to measure a certain energy of their similarity. The CC coefficient will be higher as both signals gain energy simultaneously and it is maximal when both of the compared signals are at maximum energy. Note the resulting space is always bigger than the reference space, to pad the minimal and the maximal delays, we'll expand on that later.

Until now, we discussed CC in a one dimensional data domain along the delay d. Since we want to deal with digital images (a two dimensional arrays of integers), we want to examine how CC is used in a two dimensional applications. We expand the delay d to two dimensions and sample a window area of two images – the source image and the searched template. This way the lag is the window's position in the image: the delay d of the sampling area can be used as the row and the column of the sampling area. As suggested in [2], two dimensional CC can be derived from Euclidean distance calculation:

$$d^2(I,T,u,v) = \sum_{\substack{(x,y)\in \\ \text{window}}} [\underbrace{I(x,y)}_{\substack{\text{image} \\ \text{space}}} - \underbrace{T(x-u,y-v)}_{\substack{\text{feature (template) in} \\ \text{question}}}]^2 \tag{3}$$

If we expand the right side and dismiss the sums of squares $[I(x,y)]^2$ and $[T(x-u,y-v)]^2$ as constant across the comparison window, we will find that the CC coefficient equation is:

$$\mathrm{CC}(I,T,u,v)=\sum_{\substack{(x,y)\in\\ \text{window}}}\underbrace{I(x,y)}_{\substack{\text{image}\\ \text{space}}}\cdot\underbrace{T(x-u,y-v)}_{\substack{\text{feature (template) in}\\ \text{question}}} \quad (4)$$

Here I is the source image (sized $M_x \times M_y$), T is the template (sized $N_x \times N_y$) and u, v is position of the template window. As mentioned, the CC result space is expanded to encapsulate template matching in the edges. The source image is padded with zeroes in the expanded regions as to not effect CC result.

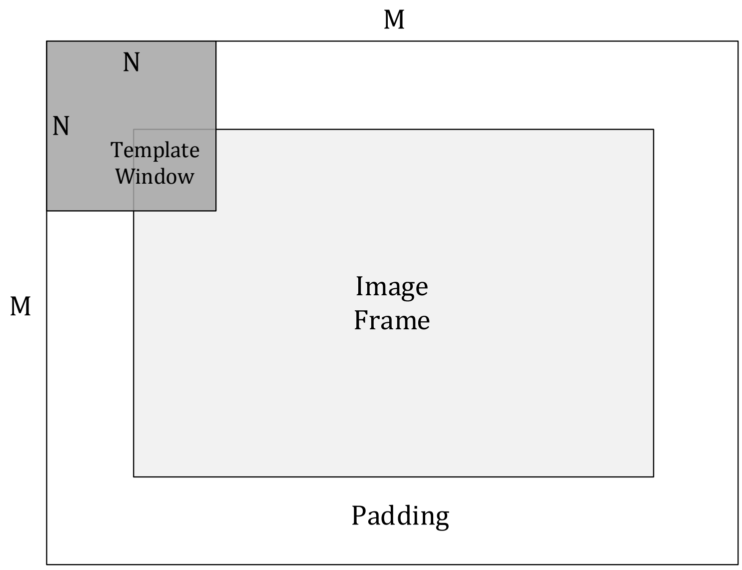


Fig. 2 Searching for a template match in an image

Note that in this manner, CC can be used in n-dimension applications. Such is one of its advantages. Another advantage is its relatively straightforward implementation, making it a considerable option for cases where quick template matching required. We'll explain such cases later on.

Using CC may be enough for a somewhat regular and relatively monotonous data, but there is a limit to its usability in a more generic setting:

- Maximal values uncertainty: a white image or a while spot will yield better CC coefficient than any true perfect match, since white pixel values are usually maximal and will yield a higher CC coefficient value
- CC is dependent on template size. I we expand the matching window, we cannot compare it to CC coefficient of the previously sized windows results, since we introduced new pixels into the sum of equation (4)
- CC is dependent on global amplitude changes in the image. If we slightly alter the brightness of the whole image by increasing pixel values, we will change the CC coefficient result

For example, let x and z be an uncorrelated 2x2 pixel windows and let y be a +1 valued window of x. Naturally, x and y yield a high correlation coefficient (with lag u=v=0). Adding a high valued pixel to window z* will affect the CC coefficient to falsely represent a high similarity with window y:

$$x = \begin{bmatrix} 1 & 2 \\ -2 & 4 \end{bmatrix} \qquad y = x+1 = \begin{bmatrix} 2 & 3 \\ -1 & 5 \end{bmatrix} \qquad z = \begin{bmatrix} 15^{*} \leftarrow 3 & 1 \\ 5 & 3 \end{bmatrix}$$

$$\mathrm{CC}(x,y,u=0,v=0)=1\cdot 2+2\cdot 3+(-2)\cdot(-1)+4\cdot 5=\mathbf{30}$$
$$\mathrm{CC}(y,z,u=0,v=0)=2\cdot 3+3\cdot 1+(-1)\cdot 5+5\cdot 3=19$$
$$\mathrm{CC}(y,z^{*},u=0,v=0)=2\cdot \mathbf{15}+3\cdot 1+(-1)\cdot 5+5\cdot 3=\mathbf{43}$$

Fig. 3 CC local extreme energy example

Thus, some kind of standardization of CC is needed. We will see a further expansion of the method that handles the issues mentioned above.

2.2. Normalized Cross-Correlation

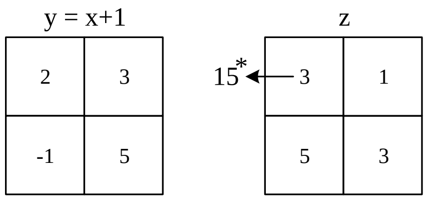
CC as given in (4) is enough for fast, general matching. As we have note, CC is not invariant to changes in global pixel values and template area size. This means that in a more demanding and comparison-heavy implementations, CC has to be normalized. In Normalized Cross Correlation, The CC coefficients are normalized by subtracting the means of relevant areas and dividing by the standard deviation. It is denoted as:

$$\mathrm{NCC}(I,T,u,v)=\frac{\sum_{(x,y)}\left(I(x,y)-\bar{I}_{u,v}\right)\left(T(x-u,y-v)-\bar{T}_{u,v}\right)}{\sqrt{\sum_{(x,y)}\left(I(x,y)-\bar{I}_{u,v}\right)^2\sum_{(x,y)}\left(T(x-u,y-v)-\bar{T}_{u,v}\right)^2}} \tag{5}$$

Where $\bar{I}_{u,v}$ and $\bar{T}_{u,v}$ are, respectively, the means of the source image I and the template T, in the watched window. It is calculated in the following manner:

$$\bar{I}_{u,v}=\frac{\sum_{x=u}^{u+N_x}\sum_{y=v}^{v+N_y} I(x,y)}{N_x \cdot N_y} \tag{6}$$

Described as above, the NCC coefficient is varying between $-1 < \mathrm{NCC}(u,v) < 1$ at any lag position in the image and it is invariable for changes to the brightness of the image and the size of the template, as these are considered using means and deviations. Let's take another look at the example in Fig. 3, this time using the normalized version of CC (The detailed calculations are elaborated in Appendix 1):



NCC(x,y ,u=0,v=0)=**1**

NCC(y,z*,u=0,v=0)= **- 0.2126**

Fig. 4 Normalized CC handling local extreme energy from Fig. 3

We can see that a local energy extreme is not producing incorrect correlation coefficient like with standard CC and the signals x and y correlate completely.

3. NCC OBSERVATIONS

The cost of Cross-Correlation is relatively efficient in its use case. Let's examine the equations (4) and (5), for a search window of size M^2 and a feature of size N^2: We run the template against the image and compare at most $(M - N + 1)^2$ reference pixels with N^2 pixels of the template. Regular CC (4) requires approximately $N^2(M - N + 1)^2$ additions and $N^2(M - N + 1)$ multiplications. For Normalized Cross-Correlation (5), all of the above holds, but the multiplications cost is $M^2 + N^2$, and we also perform $(M - N + 1)^2$ squarings.

Normalized Cross-Correlation is widely used in image registration domain. Barnea and Silverman [3] suggested extensive optimizations for NCC and categorized it, among other detection algorithms, as a 'SSDA' (Sequential Similarity Detection Algorithms) class algorithm. In [2], J. P. Lewis presented the faster, pre-calculated version of NCC with use of Fourier transform for the numerator of (5), reducing additions cost to $4(M - N + 1)^2$. Briechle and Hanebeck have further expanded Lewis' NCC with Sum Expansion [4].

3.1. Template Matching Example

We have experimented with NCC under the MATLAB environment. We present here a result for a simple sub-image registration using NCC (Based on [5]). The image and the template are:

Fig. 5 Tested image and the sub-pattern

In this test, we run the template against the source image, calculating the maximal NCC coefficient across the image. A result is a map of NCC values calculated across the comparison space (Fig. 6). If the maximal coefficient value is above a certain threshold, we consider it a match. We paint the template over the source image, centered at the maximal NCC coefficient. The full MATLAB code for the test is provided in Appendix 2.

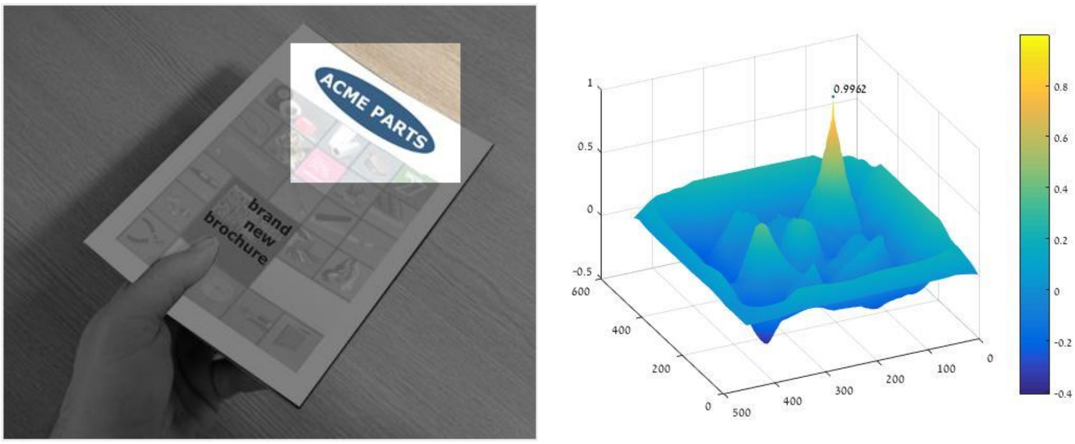


Fig. 6 Matching test result and a plot of the NCC coefficients map

3.2. Normalization Advantages

To demonstrate NCC invariance to global changes to energy values, we brighten the template image by increasing each pixels value by an equal amount.

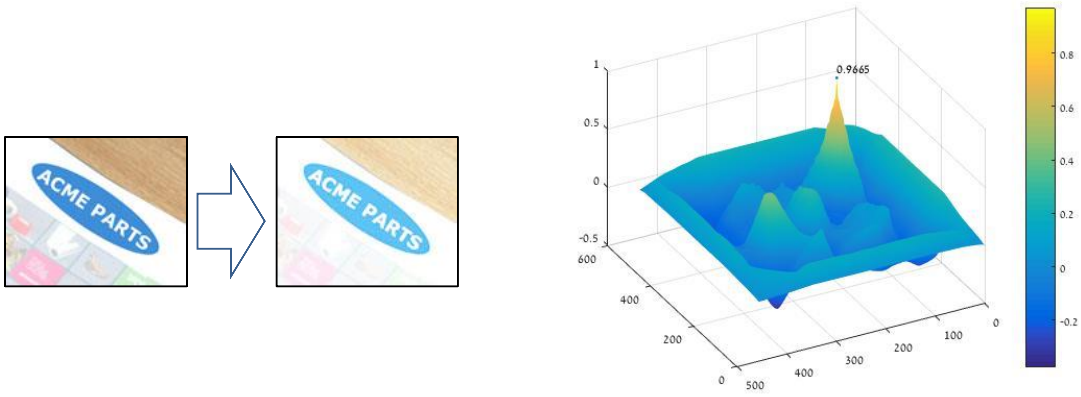


Fig. 7 NCC result with brightened template

Although the match is found, note that the resulted maximal NCC coefficient is slightly lower than that of a non-brightened template. This is explained by the presence of white (maximal valued) pixels in the template window.

Let's see how NCC handles changes in window size: we expand the template window by about 15 pixels. The scale is not changed, but the template now simply represents a bigger sub-slice from the source image. The result is correct and proves NCC's invariability to template size.

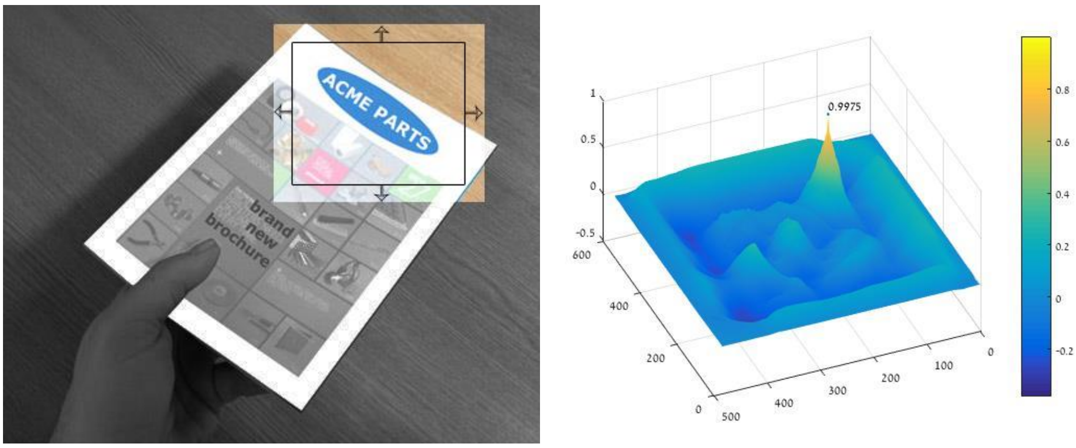


Fig. 8 Increasing sub-pattern size in NCC

In the simple cases above, the result is straightforward. But what happens when we introduce some of the more demanding changes that are naturally met in image processing and computer vision?

3.3. Multiple patterns presence

We have introduced several similar sub-pattern to the source image: We have pasted a copy of the "Acme Parts" title in the brochure to see how the NCC algorithm reacts to it.

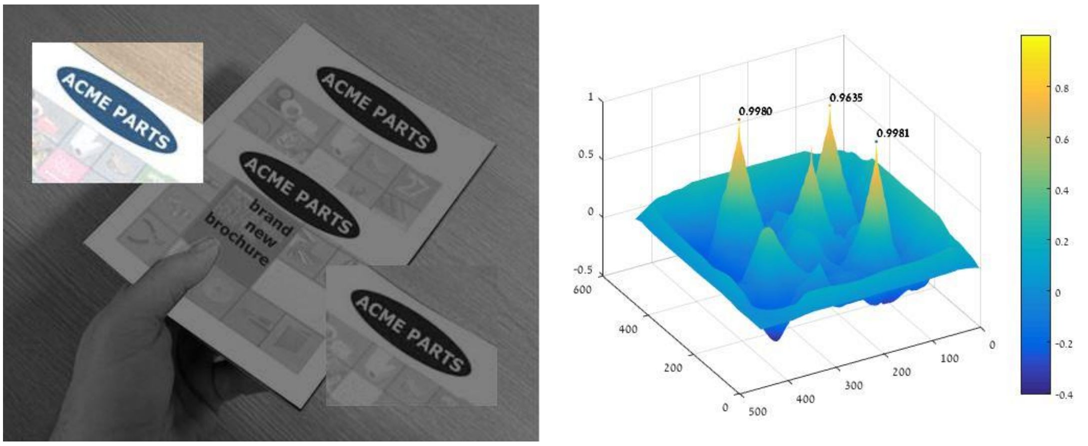


Fig. 9 Multiple sub-pattern matches present in NCC

The resulted map indeed shows a high NCC coefficient where expected. Being a serial (SSDA) kind of an algorithm, NCC has selected the first occurrence a match. In this case, we are not guaranteed that the match is indeed the precise pattern we are looking for. Those cases prove NCC at a disadvantage when multiple similar sub-templates present in an image (Mosaic, bricks, textures). For those uses, a constant change in the NCC coefficient threshold for detection may be required, making NCC less convenient.

3.4. Scale Variance

Another interesting case is where the searched template is larger in scale than the source image (or alternatively, where the source image is scaled down). In such case, no match is found. The scale difference makes the lag stepping of the NCC inconsistent, resulting in an incoherent coefficients map.

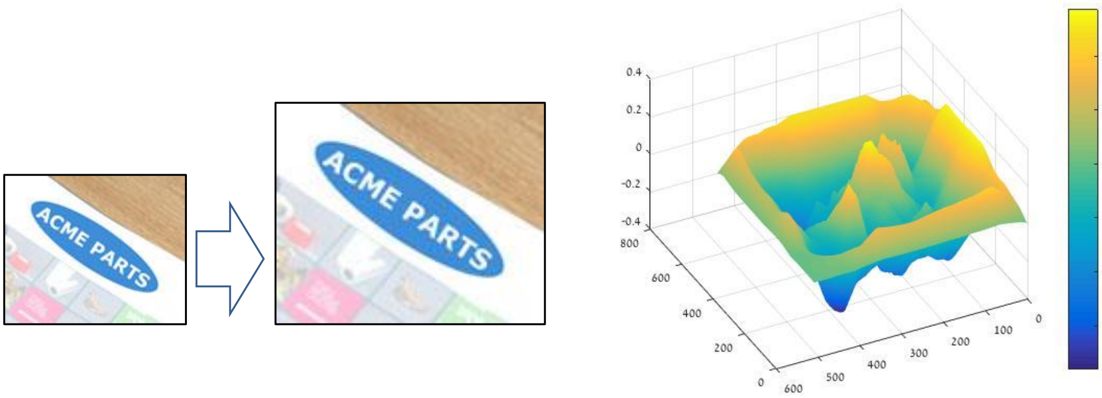


Fig. 10 The effect of scale variation on NCC

3.5. Rotational Variance

We tested NCC behavior with rotated pattern window. The rotation margins are padded with black (zero valued) pixels to negate their effect on the calculation. The match is not found and we are left with an uncomprehensive coefficients values map.

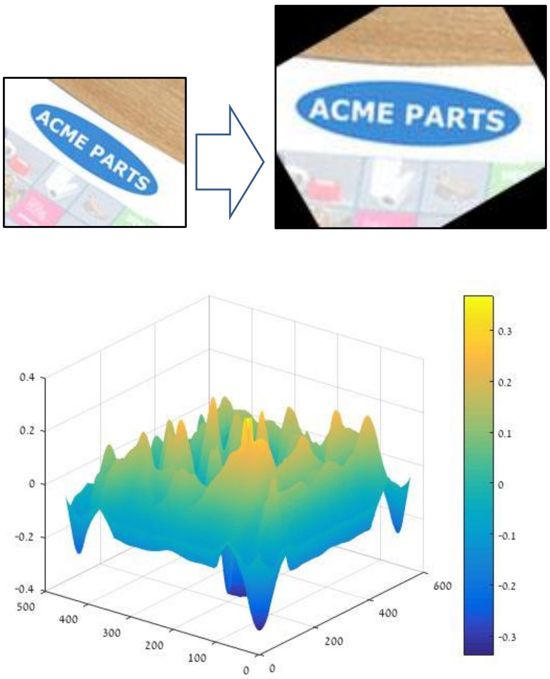


Fig. 11 The effect of angular variation on NCC

This implies NCC invariability to various geometric distortion in the image space. An important example of such deformation is the change in perspective, which is often met in digital photography. This disadvantage makes NCC unfavorable candidate in certain cases, such as template matching over a shaky video flow.

4. CONCLUSION

We have introduced the Cross-Correlation method and its basic principles in signal processing, as a mathematical tool for similarity representation. We have seen how we can expand CC to be used in multiple dimensioned data spaces, we expand the lag between two signals to a lag between two pixel windows to make CC applicable for template matching in computer vision.

As the limitations of the standard CC limits its convenience when dealing with frequent obstacles in sub-pattern registration, we have introduced and elaborated on Normalized version of CC, as a means to standardize the Correlation coefficient. We demonstrated the increased robustness of the NCC and ran various testing in MATLAB environment to learn the advantages and the limitations of the NCC.

The NCC remains one of the prominent techniques in pattern registration due to its advantages we have demonstrated those advantages:

- Straightforward implementation, low computational costs
- Invariability to template window size
- Invariability local energy extremes (high valued pixels, errors)
- Invariability to global energy changes (brightness shift)

NCC is not perfect. We have tested it over various demanding cases usually encountered in digital imaging and found the cases in which the basic NCC is at a disadvantage:

- Uncertainty with multiple sub-pattern presence
- Sensitivity to image scale changes
- Sensitivity to rotation of the frame
- Sensitivity to perspective distortions

Solutions that enhance NCC or use it as a component were suggested, to address the above issues. Nevertheless, whether enhanced or not, NCC is a reasonable choice for a base pattern registration method in many applications. Also, as presented in [6], NCC is among the best performers in its class.

5. Appendix 1

Calculation of NCC coefficients of the example in Fig. 4 using the formula (5):

x

1	2
-2	4

y = x+1

2	3
-1	5

z

3	1
5	3

15* ← 3

$$\mathrm{NCC}(I,T,u,v)=\frac{\sum_{(x,y)}\left(I(x,y)-\bar{I}_{u,v}\right)\left(T(x-u,y-v)-\bar{T}_{u,v}\right)}{\sqrt{\sum_{(x,y)}\left(I(x,y)-\bar{I}_{u,v}\right)^2\sum_{(x,y)}\left(T(x-u,y-v)-\bar{T}_{u,v}\right)^2}},\quad \bar{I}_{u,v}=\frac{\sum_{x=u}^{u+N_x}\sum_{y=v}^{v+N_y}I(x,y)}{N_x\cdot N_y}$$

$$\bar{x}_{u,v}=\frac{5}{2\cdot 2}=1.25=\frac{5}{4};\ \bar{y}_{u,v}=\frac{9}{2\cdot 2}=2.25=\frac{9}{4};\ \overline{z*}_{u,v}=\frac{24}{2\cdot 2}=6$$

$$\sum_{(x,y)}\left(\mathrm{x}(x,y)-\bar{x}_{u,v}\right)^2=(1-\frac{5}{4})^2+(2-\frac{5}{4})^2+(-2-\frac{5}{4})^2+(4-\frac{5}{4})^2=$$

$$=(-\frac{1}{4})^2+(\frac{3}{4})^2+(-\frac{13}{4})^2+(\frac{11}{4})^2=\frac{1}{16}[(1)+(9)+(169)+(121)]=\frac{300}{16}=\frac{75}{4}=18.75$$

$$\sum_{(x,y)}\left(y(x,y)-\bar{y}_{u,v}\right)^2=(2-\frac{9}{4})^2+(3-\frac{9}{4})^2+(-1-\frac{9}{4})^2+(5-\frac{9}{4})^2=...=18.75$$

$$\sum_{(x,y)}\left(z*(x,y)-\overline{z*}_{u,v}\right)^2=(15-6)^2+(1-6)^2+(5-6)^2+(3-6)^2=(121+25+1+9)=156$$

$$\text{NCC(x,y,u=0,v=0)=}\frac{(1\text{-}1.25)\cdot(2\text{-}2.25)+(2\text{-}1.25)\cdot(3\text{-}2.25)+(\text{-}2\text{-}1.25)\cdot(\text{-}1\text{-}2.25)+(4\text{-}1.25)\cdot(5\text{-}2.25)}{\sqrt{\sum_{(x,y)}\left(\mathrm{x}(x,y)-\bar{x}_{u,v}\right)^2\cdot\sum_{(x,y)}\left(y(x,y)-\bar{y}_{u,v}\right)^2}}=$$

$$=\frac{(0.0625)+(0.5625)+(10.5625)+(7.5625)}{\sqrt{18.75\cdot 18.75}}=\frac{18.75}{18.75}=\mathbf{1}$$

$$\text{NCC(y,z*,u=0,v=0)=}\frac{(\text{-}0.25)\cdot(11)+(0.75)\cdot(\text{-}5)+(\text{-}3.25)\cdot(\text{-}1)+(2.75)\cdot(\text{-}3)}{\sqrt{\sum_{(x,y)}\left(y(x,y)-\bar{y}_{u,v}\right)^2\cdot\sum_{(x,y)}\left(z*(x,y)-\overline{z*}_{u,v}\right)^2}}=$$

$$=\frac{(\text{-}2.75)+(\text{-}3.75)+3.25+(\text{-}8.25)}{\sqrt{18.75\cdot 156}}=\frac{11.5}{15\sqrt{13}}=\mathbf{-0.2126}$$

6. Appendix 2

Feature search NCC test in MATLAB code implementation

```
%read the images
brochure    = imread('brochure.jpg');
text        = imread('brochure_text.jpg');  % sub-image cropped from 'brochure.jpg'

figure('Name', 'Brochure orig');
imshow(brochure)
figure('Name', 'Sub-Image');
imshow(text)

%calculate the Normalized Correlation Coefficients
ncc = normxcorr2(text(:,:,1),brochure(:,:,1));      % use 'text' as a template
figure('Name', 'Correlation Result Graph');
surf(ncc), shading FLAT
colorbar;                                    %show a color bar along the 3D graph

%find the maximal NCCed value and it's index
[max_val, max_idx] = max(abs(ncc(:)));       %the resulted 'max_idx' is linear
[idx_row, idx_col] = ind2sub(size(ncc),max_idx(1)); %get the index of the max NCC value
%cut off the correlation padding frame
xoffset = idx_col - size(text,2);
yoffset = idx_row - size(text,1);

%display the sub-image: first paste it in the original image
padded = uint8(zeros(size(brochure)));
padded(round(yoffset+1):round(yoffset+size(text,1)),round(xoffset+1):round(xoffset+
size(text,2)),:) = text;


figure('Name', 'Found Sub-Image');
imshowpair(brochure(:,:,1),padded,'blend')
```

7. REFERENCES

[1] P. Bourke, "http://paulbourke.net/miscellaneous/correlate/," 1996. [Online]. [Accessed 11 2014].

[2] J. P. Lewis, "Fast Normalized Cross-Correlation," *Vision Interface,* pp. 120-123, 1995.

[3] D. I. Barnea and H. F. Silverman, "A Class of Algorithms for Fast Digital Image Registration," *IEEE Trans. Computers,* vol. 21, pp. 179-186, 1972.

[4] K. Briechle and U. D. Hanebeck, "Template Matching Using Fast Normalized Cross Correlation," *Optical Pattern Recognition XII,* 2001.

[5] J. Sarvaiya, S. Patnaik and S. Bombaywala, "Image Registration by Template Matching Using Normalized Cross-Correlation," *International Conference on Advances in Computing, Control, & Telecommunication Technologies,* pp. 819-822, 2009.

[6] P. J. Burt, C. Yen and X. Xu, "Local Correlation Measures for Motion Analysis: a Comparative Study," *IEEE Conf. Pattern Recognition Image Processing,* pp. 269-274, 1982.

[7] D. Scharstein and R. Szeliski, "vision.middlebury.edu/stereo/," [Online]. [Accessed 12 2014].

[8] R. Nevatia, "Depth Measurement by Motion Stereo," *Computer Graphics And Image Processing 5,* pp. 203-214, 1976.

8. FIGURES INDEX

Fig. 1 Visualization of Cross-Correlation between two signals 4
Fig. 2 Searching for a template match in an image 5
Fig. 3 CC local extreme energy example 6
Fig. 4 Normalized CC handling local extreme energy from Fig. 3 7
Fig. 5 Tested image and the sub-pattern 8
Fig. 6 Matching test result and a plot of the NCC coefficients map 8
Fig. 7 NCC result with brightened template 9
Fig. 8 Increasing sub-pattern size in NCC 9
Fig. 9 Multiple sub-pattern matches present in NCC 10
Fig. 10 The effect of scale variation on NCC 10
Fig. 11 The effect of angular variation on NCC 11

HAVE A GREAT DAY!