Template Matching in Images

Literature Review

Jonathon Everatt
Department of Computer Science
University of Cape Town
Cape Town, Western Cape, South Africa
everatt.jonathon@gmail.com

Abstract

This paper investigates different approaches to Template Matching in images and their relative strengths and weaknesses to determine if these approaches can be used in an upcoming project on planting patterns in orchard/crop management. Popular similarity measures are examined both of a simple and complex nature. SSD, SAD, NCC and ZNCC are presented due to the accuracy and low computational cost. The BBS and DDS measures are used for complex cases where the template may be distorted or occluded. This paper also examines the computational efficiency of template matching how it is parallelizable and can achieve speed up with GPU implementation. The problems of rotation and scale and some of the solutions to these issues are investigated both in general terms and if applicable to algorithms discussed in the paper. This paper discusses how these techniques are applicable to the upcoming project on planting patterns and how we can use the previous work done in the field for tree identification.

CCS Concepts

Template Matching, Computer Vision, Image Processing, Cauchy Schwarz.

Keywords

Template Matching, Sum of Squared Difference (SSD), Sum of Absolute Difference (SAD), Normalized Cross Correlation (NCC), Zero-means Normalized Cross Correlation (ZNCC), Full Search (FS), Best Buddies Similarity (BBS), Deformable Diversity Similarity (DDS), Best Buddies Pairs (BBP), Cauchy Schwarz Inequality

Introduction

The focus of the upcoming project is to identify planting patterns for crop/orchard management. As these have a major effect on the final yield of the crops for the farmer as different advantages for the farm such as effective cross-pollination, efficient irrigation, and accessibility. Farmers can also have multiple planting patterns on the same plantation and variations within the same planting pattern. These variations can stem from intra-pair spacing, inter-pair spacing and row spacing.

This literature review will conduct a critical review of template matching techniques in images and how it can be applied to aerial footage of plantations to aid in crop/orchid management.

Template matching in Image Processing and Computer Vision is the process of determining whether predetermined sub-image(s) are present within a larger source image(s) and the approximate location of those templates [1]. In this case the templates would comprise the planting patterns within the images of the aerial footage of plantations or template matching could be used to identify the trees in the image [2]. The use of template matching to identify trees from aerial footage has already been successfully implemented [2,6,7,8]. A naïve approach to template matching which implements a Full Search (FS) algorithm can become extremely computationally expensive [3].

Template matching can present a solution to the project in two ways. One, template matching can be used to identify the trees in the orchid/plantation. This way of using template matching has already been implementing and has been proven to work with an acceptable accuracy range [2,6,7,8]. Zainuddin and Daliman [2] encountered a large problem in their study of misidentifying non-Rubber trees as Rubber trees but managed 74,03% accuracy on identifying Rubber trees. Our project will not suffer from issues like these as we are not trying to identify by species, rather just the fact that there is a Tree there. Two, template matching can also be used to identify the planting patterns. This could be used made efficient by turning the image into a grayscale (single channel) image [3,21]. Although the classic problems of template matching, namely that it is not good at dealing with different rotations and scale, will have to be solved for it to be of effective use here. This review will analyse existing solutions to each problem later in the paper to see if this could be an effective solution.

Three basic matching techniques

In Image matching there are several techniques that can be used to solve the problem. The 3 main techniques for matching images are Intensity-based matching, feature-based matching, and relational matching. These techniques are categories in which the template matching algorithms and similarity measures, that the is paper will review, are categorized by.

Intensity-based: Also called area-based techniques. Example methods in this area is NCC, ZNCC, SAD, and SSD. This method measures the pixel of a sub image to an area of the image. [4,5,16]

Feature-based: These techniques such as DDS and BBS extract features for detection from the template and detect them on the image. [4,5,16]

Relational: This approach uses relations between the features of the image to determine a match with the sub image. These techniques are generally less accurate but more robust than others [4,16]

Template Matching Similarity Measure

The accuracy and computational cost of a template matching algorithm is dependent on the similarity measure equation(s) it uses to calculate the difference between the image and the template [12,11].

The most popular template matching methods are Sum of Squared Difference (SSD), Sum of Absolute Difference (SAD) and Normalized Cross Correlation (NCC) [17]. Of these methods the first two are distortion methods and the last is a correlation method. Distortion methods calculate of the minimum distorsion between the image and template like sum of squared difference and the sum of absolute difference. Correlation methods calculate off the maximum correlation like normalized cross correlation [14, 9]. The two distorsion methods are FS algorithms whereas NCC is an exhaustive algorithm, although due to the difference in the amount calculation necessary the distorsion methods are still more computationally efficient [9,13,20,22]

Normalized cross correlation is more robust in measuring similarity between the template and the image but due to it is multiple uses of multiplication it is also more computationally expensive [10,11]. Both methods are very computationally efficient, although they do not handle complex cases very well and cannot fail to find templates are result in an incorrect match [12]. For more complex cases of Template matching, more complex algorithms using different similarity measures Best-Buddies Similarity [11] and the Deformable Diversity Similarity [12]. The complex cases tend to be more robust and have a better ability to deal with the issues of template matching such as occlusion, but the price is paid in computational expense [12,11]

Sum of Squared Difference (SSD)

This algorithm is not very robust but extremely fast computationally. The simplicity of the measure that gives the algorithm its speed does so at the cost of robustness. Scale, rotation, and complexity changes from the template would not be picked up at all by this measure and would lead to a false match or no match.[22] A big issue that is faced by simple algorithms that measure on a pixel level is that they punish pixel differences quite heavily even when it is a decent match. This is prevalent when there are illumination changes between the template and image. Can produce a significant difference in the pixel values which can lead to a rejection of a template that should have been a good match. [11,3]

$$SSD(x,y) = \sum_{(x,y)\in I}(I_1[x,y] - I_2[x,y])^2 \quad (1)$$

Although equation 1 seems complex this is simply the computation for that is used by the computer. It is still only calculating the squared difference of pixels between the template and image [10,22]

The SSD similarity measure is a pixel-by-pixel measure that calculates the intensity difference between each of the pixels. The matching point can be found by searching for the minimum value in the image matrices as that will be the point where there is the smallest intensity difference between the template and image. [13,15] Due to the popularity of the measure there were initially FS methods and exhaustive methods have also been developed [22].

Sum of Absolute Difference (SAD)

The SAD similarity measure is a pixel-based measure that takes the absolute difference between two pixels in a single dimension.

$$SAD(x,y) = \sum_{(x,y)\in I}|I_1[x,y] - I_2[x,y]| \quad (2)$$

Equation 2 is a general equation for calculation the SAD measure but there are multiple versions that can calculate the sum of absolute differences between 1 dimension, 2 dimensions, functions, and matrices.[20]

This distortion method performs a FS of the image and take the lowest value as it is matching point. It is very similar to the SSD measure in many ways except the calculation. [21,5]

An optimized SAD (OSAD) algorithm was proposed for fast template matching for facial recognition by Dawoud et al [20]. Through an optimized equation (equation 3) they managed to achieve fast and accurate results for finding a face in an image. This, in general terms, was done by creating a generic face template from many faces, divide the template into squares which are then matched with the image using the above equation. Once this was completed the algorithm would select the minimum value from the image matrix and that is the face. Another benefit of this approach apart from its high accuracy (96% in the tests conducted) is that it is robust against illumination changes which area-based approached normally are not. [19, 20,3].

$$OSAD(A,B) = \sum_i \sum_j \frac{|A(i,j) - B(i,j)|}{\max(A(i,j), B(i,j))} \quad (3)$$

Normalized Cross Correlation (NCC)

The NCC similarity measure is more robust than the SSD measure but carries with it more mathematical computation and is therefore more expensive [10]. The cost of this extra computation can also be calculated using the Cauchy Schwarz inequality [16]. The values used in the NCC algorithm is normalized to account for intensity changes between the template and image and could produce a false match. This algorithm searches for a match by searching for a maximum value in the image matrices after running the algorithm, which would evidently be the highest correlation coefficient [16].

Although Normalized Cross Correlation has been shown to be efficient and easy to implement the problem arises that the only parameters are two shifts between the template and image and cannot deal with rotation or scale changes. [4] Furthermore even a template that should perfectly match and get a correlation coefficient of 1 at best could only get 0.99 and more likely get lower, more in the region of 0.95. Though this issue is easily avoidable as we just set a bound for what is considered an acceptable match [9]. The NCC algorithm is highly popular due to its robustness when compared to the previous two algorithms while still maintaining a computational advantage over algorithms like BBS that can deal with the complex cases that this algorithm cannot deal with. [11,10]

$$\boldsymbol{\gamma} = \frac{\sum_{x,y}(f(x,y) - f'u,v)(t(x-u,y-v) - t')}{\sqrt{\sum_{x,y}(f(x,y) - f'u,v)^2 \sum_{x,y}(t(x-u,y-v) - t')^2}} \quad (4)$$

The equation above a basic definition for a normalized cross correlation coefficient. [23]

Zero means Cross Correlation (ZNCC)

ZNCC is a similarity measure for a template matching technique that was initially developed for grayscale images that is exhaustive and guarantees to find a global maximum of the correlation coefficient [3,21]. This was then expanded upon to be applicable to multi-channel images. The proposed algorithm by Mattoccia et al proved to have speed for ZNCC in multi-channel images when compared to its FS variant for all image sizes and efficiency increases with size [3].

$$ZNCC = \frac{\sum_{(u,v)\in I} I_1[u,v] \cdot I_2[u,v]}{\sqrt{\sum_{(u,v)\in I} I_1^2[u,v] \cdot \sum_{(u,v)\in I} I_2^2[u,v]}} \quad (5)$$

Equation 5 [21] shows a general ZNCC definition that can be used on single channel images. There is another version developed for multichannel image [3].

The ZNCC algorithm returns coefficients from -1 to 1, with 1 being a perfect match and -3 being the negative of the template. (Typically, above 0.8 is considered a decent match). The equation below computes the correlation coefficient for the template. The computational expense of this algorithm compared to SSD and SAD is clear just based on the differences in their equations.[15]

Best Buddies Similarity

The Best Buddies Similarity uses a novel approach to template matching. Rather than matching every pixel and calculating the difference or correlation between the template and image. This similarity compares key features that have been analysed from the templates to match with the image. This similarity measure is based on the Nearest-Neighbour as it matches template features with target features. The features that are matched are called *Best Buddy pairs* (BBP) and are only considered a pair if each point is the nearest neighbour of the corresponding point. Due to only sampling a small number of points, this algorithm is very robust against outliers and noise and the BBS between two points from the same distribution is maximal [12,11].

This presents us a with a robust, parameter-free algorithm that is far more resistant to occlusion that the previous "simple" approaches to template matching. Though the "simple" approaches are still very popular and efficient so when complexity is not an issue those algorithms are preferable because of their much faster run times [12,18].

This algorithm is very useful and solves the template matching issue for complex cases far better than the previous similarity measure that have been described. There are still drawbacks to this: First is the computation time which increases with image size [12], then there is the problem of rotation that this cannot deal with and finally the problem of scale can also not be dealt with by this measure either [18]. Solutions to the first two issues will be discussed in this paper but the solution to scale can come from solving the run time issue. As it will then be possible to rerun the algorithm multiple times with different sized templates [9].

Deformable Diversity Similarity

DDS is a similarity measure that was developed to improve upon the BBS measure by reducing the computational. This algorithm uses "The diversity of Nearest Neighbour (NN) matches between template points and target points is indicative of similarity between them. The deformation implied by the NN field should be explicitly accounted for". The major difference between this metric and the BBS metric is the calculation it uses for its similarity and the penalty on spatial deformation. [11]

In figure 1, we can see that the BBS algorithm increases in run time as the template and image size increases but the DDS measure remains constant for all sizes measure. This suggests that this algorithm is very useful for dealing with the more complex cases. Except it has a drawback in that it cannot deal with changes to scale.

The paper suggests the way to overcome this would be to map over multiple scales but have not proved it to be possible. [11]

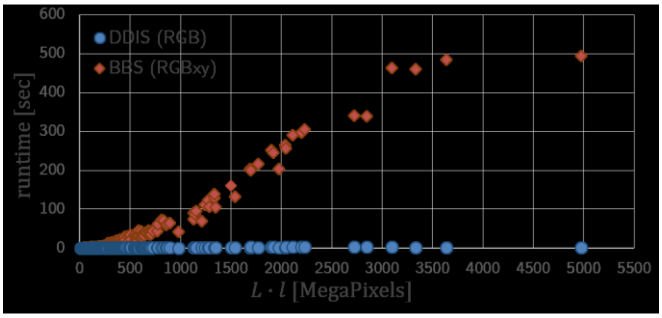

Figure 1: Runtime of 270 image pairs for DDS and BBS measures [11]

Efficiency

As stated previously template matching, especially a naïve approach implementing a simple similarity difference equation such as NCC, SSD or SAD can be extremely computationally expensive when run on a CPU. These simple measures can be parallelized and implemented onto a GPU for speed up [3,9]. These simple measures with a naïve implementation perform a Full Search (FS). This means that they compare every possible position of a template on an image. Although speed up is possible this wastes a large amount of computational effort. Rather it is better to implement this measure by using an exhaustive search, this essentially searches the same amount of viable template spaces on the image without the wasted computation [13]. The ability to implement an exhaustive algorithm instead of a FS can result in major runtime improvements especially of large images.

TABLE I
PROCESSING TIMES OF MATCHING MULTIPLE IMAGES

#	Step size	CPU runtime (ms)	GPU runtime (ms)	CPU time / GPU time
1	-	341.3	24.1	14.1
2	0.9	621.5	38.2	16.3
3	0.6	909.6	56.5	16.1
4	0.45	1170.2	71.6	16.3
5	0.36	1439.2	91.1	15.8
10	0.18	2859.3	178.2	16.1
15	0.12	4268.5	264	16.2
20	0.09	5621.3	350.1	16.1
25	0.072	7078.1	437.9	16.2
30	0.06	8479.3	523.5	16.2
35	0.051	9855.2	609.8	16.2
40	0.045	11242	695	16.2
45	0.04	12629.3	784	16.1
50	0.036	14041.3	871.3	16.1
55	0.032	15454.7	955.5	16.2
60	0.03	16894.4	1041.6	16.2

Figure 2 CPU vs GPU runtimes [3]

Rotation in Template Matching

One of the major issues that template matching algorithm are sensible to (especially the simple approaches) is rotation. Since the template it matched from a single orientation to the image and often the ad hoc solution that is used when it is necessary to deal with rotation is to develop new templates that are just rotations of the original. This is not a robust method for dealing with rotation and so it is necessary to come up with a robust way of dealing with it. [3,18**]**

Template matching is an algorithm that does not inherently deal with differences in rotation and scale between the source image and the template [4,5,9,18]. One solution is to implement a two-stage template matching algorithm that to deal with rotation in template and source image. The initial stage will use a sub-template to determine the area most likely to contain the template using a similarity measure. Followed by the second stage which shifts the templates until the highest similarity measure could be found [1].

Using Best Buddies Pairs (BBP) from the BBS measure, an algorithm proposed by Yang, Chen, and Li [18] uses these pairs to calculate the angle of rotation for the template. The original BBS measure is not able to deal with large angle variance, however with this added calculation on the image and in the algorithm the template can match to the image with large angle changes. There is a computational cost for this extra processing it, but it comes with increased accuracy. Other methods such as rotational NCC and SIFT have faster computation at the cost of accuracy. Although BBS time increases with image size, exponentially this shows that algorithms based on BBS can be developed to deal with rotation in images and DDS has proposed a solution to the computational cost based on the BBS measure [11,12,18].

Scale in Template Matching

The second problem for template matching algorithms is changes in scale between the template and the image. The reviewed measures in this paper all are not able to account for scale. While the complex algorithms can build upon BBS to deal with rotation by orientating the template in the correct position this style of solution is not viable for scale changes [18].

One solution to changes in scale proposed by Gabor Kertesz, Sandor Szenasi and Zoltan Vamossy is to generate multi-scale versions of the template and then run the algorithm. By parallelizing the algorithm and implementing it on a GPU it is possible to make the algorithm run-time realistic. The drawbacks to this are that the proposed method is not able to be used on the more complex methods, or at least not shown to be able to be used on them in the paper. The results of parallelizing these naïve implementations of similarity measures showed a 16x speed up when implemented on the GPU without loss of accuracy due to parallelization. [9]

Discussion

This paper has reviewed simple and complex approaches to template matching and their relative advantages and disadvantages as well as solutions to the classic problems of rotations and scale that template matching algorithms face. For implementation in the project template matching could be used to identify trees and planting patterns. The method for matching trees is already viable as has been found by [2,6,7,8] and the issues faced in these papers such as misidentifying tree type is not an issue for the project. The planting patterns could also be found by this algorithm by making a lot of templates for them or using a similarity measure like BBS for complex patterns.

Template matching is a powerful image processing technique that can efficiently find and classify whether a sub image is in in a larger image. Using trees and/or patterns as a template this technique will be useful going forward in the project.

Conclusions

This paper has shown several simple and complex techniques for template matching and their relative advantages and disadvantages. The main question that needs to be asked when deciding on a template matching algorithm is the complexity needed to match the template and if there are any computational limits.

The classic problems of scale and rotation have also been investigated and possible solution have been proposed for both. Rotation is the lesser of the two problems as there are algorithms that can incorporate it at a computational cost. Though the issue of scale can also be overcome by making multi-scale templates, this has its own computational cost but if the algorithm is fast enough such as the simple implementations of DDS, then it is a solution.

Finally in the context of the project we explored ways in which template matching has been used to map trees of images and possibly if these techniques could be incorporated for detecting planting patterns.

References

[1] GOSHTASBY A. 1985. Template matching in rotated images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (3), 338-344.

[2] ZAINUDDIN, N. E., AND DALIMAN, S. 2020. Analysis of Rubber Tree Recognition Based on Drone Images. In *IOP Conference Series: Earth and Environmental Science* (Vol. 549, No. 1, p. 012012). IOP Publishing.

[3] MATTOCCIA, S., TOMBARI, F., AND DI STEFANO, L. 2011. Efficient template matching for multi-channel images. *Pattern Recognition Letters*, *32*(5), 694-700.

[4] SWAROOP, P., AND SHARMA, N. 2016. An overview of various template matching methodologies in image processing. *International Journal of Computer Applications*, *153*(10), 8-14.

[5] GRUEN, A. 2012. Development and status of image matching in photogrammetry. *The Photogrammetric Record*, *27*(137), 36-57.

[6] NORZAKI, N., AND TAHAR, K. N. 2019. A comparative study of template matching, ISO cluster segmentation, and tree canopy segmentation for homogeneous tree counting. *International Journal of Remote Sensing*, *40*(19), 7477-7499.

[7] ISIP, M. F., CAMASO, E. E., DAMIAN, G. B., AND ALBERTO, R. T. 2018. Estimation of Mango Tree Count and Crown Cover Delineation using Template Matching Algorithm. *International Journal for Research in Applied Science and Engineering Technology*, *6*(3), 1955-1960.

[8] KALANTAR, B., MANSOR, S. B., SHAFRI, H. Z. M., AND HALIN, A. A. 2016. Integration of template matching and object-based image analysis for semi-automatic oil palm tree counting in UAV images. In *Proceedings of the 37th Asian Conference on Remote Sensing, Colombo, Sri Lanka* pp. 17-21.

[9] KERTÉSZ, G., SZÉNÁSI, S., AND VÁMOSSY, Z. 2015. Performance measurement of a general multi-scale template matching method. In *2015 IEEE 19th International Conference on Intelligent Engineering Systems (INES)* pp. 153-157. IEEE.

[10] HISHAM, M. B., YAAKOB, S. N., RAOF, R. A., AND NAZREN, A. A. 2015. Template matching using sum of squared difference and normalized cross correlation. In *2015 IEEE student conference on research and development (SCOReD)* pp. 100-104. IEEE.

[11] DEKEL, T., ORON, S., RUBINSTEIN, M., AVIDAN, S., AND FREEMAN, W. T. 2015. Best-buddies similarity for robust template matching. In *Proceedings of the IEEE conference on computer vision and pattern recognition* pp. 2021-2029.

[12] TALMI, I., MECHREZ, R., AND ZELNIK-MANOR, L. 2017. Template matching with deformable diversity similarity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* pp. 175-183.

[13] OUYANG, W., TOMBARI, F., MATTOCCIA, S., DI STEFANO, L., AND CHAM, W. K. 2011. Performance evaluation of full search equivalent pattern matching algorithms. *IEEE transactions on pattern analysis and machine intelligence*, *34*(1), 127-143.

[15] L DI STEFANO AND S MATTOCCIA. 2003, “Fast Template Matching using Bounded Partial Correlation,” Machine Vision and Applications, vol. 13, pp. 213–221

[16] DI STEFANO, L., AND MATTOCCIA, S. 2003, September. A sufficient condition based on the Cauchy-Schwarz inequality for efficient template matching. In *Proceedings 2003 International Conference on Image Processing (Cat. No. 03CH37429)* Vol. 1, pp. I-269. IEEE.

[17] Wei, S. D., AND Lai, S. H. 2008. Fast template matching based on normalized cross correlation with adaptive multilevel winner update. *IEEE Transactions on Image Processing*, *17*(11), 2227-2235.

[18] YANG, Y., CHEN, Z., LI, X., GUAN, W., ZHONG, D., AND XU, M. 2020. Robust template matching with large angle localization. *Neurocomputing*, *398*, 495-504.

[19] ATALLAH, M. J. 2001. Faster image template matching in the sum of the absolute value of differences measure. *IEEE Transactions on image processing*, *10*(4), 659-663. [20] OSAD paper

[21] MATTOCCIA, S., TOMBARI, F., AND DI STEFANO, L. 2008. Fast full-search equivalent template matching by enhanced bounded correlation. *IEEE transactions on image processing*, *17*(4), 528-538.

[22] SIDIA, W. D., AND WIBAWAA, I. G. A. Sum of Squared Difference (SSD) Template Matching Testing on Writing Learning Application. Jurnal Elektronik Ilmu Komputer Udayana p-ISSN, 2301, 5373.

[23] BRIECHLE, K., & HANEBECK, U. D. 2001. Template matching using fast normalized cross correlation. In Optical Pattern Recognition XII (Vol. 4387, pp. 95-102). International Society for Optics and Photonics.