Fast Affine Template Matching over Galois Field

Chao Zhang zhang@scv.cis.iwate-u.ac.jp Takuya Akashi akashi@iwate-u.ac.jp

In this paper, we address the problem of template matching under affine transformations with general images. Our approach is to search an approximate affine transformation over a binary Galois field. The benefit is that we can avoid matching with huge amount of potential transformations, because they are discretely sampled. However, a Galois field of affine transformation can still be impractical for exhaustive searching. To approach the optimum solution efficiently, we introduce a level-wise adaptive sampling (LAS) method under genetic algorithm framework. In LAS, individuals converge to the global optimum according to a levelwise selection and crossover while the population number is decreased by a population bounding scheme. Specifically, our paper is to infer an approximate affine transformation \hat{T} from a given candidate set. In the best case, \hat{T} equals to transformation \overline{T} . \overline{T} is the closest transformation to ground truth T' among all the candidate transformations. An natural way to estimate \hat{T} is to minimize SAD. Formally, our purpose can be denoted as:

$$\hat{T} = \underset{T \in \mathbb{F}_{2^{6n}}}{\operatorname{arg\,min}} S(I_1, I_2, T). \tag{1}$$

According to [1], a general affine transformation matrix can be decomposed into $T = TrR_2SR_1$, where R_1 represents matrix operation of 1st rotation, *S* is scale operation in x-axis and y-axis, R_2 is 2nd rotation, Tr is translation operation in x-axis and y-axis. Transformations over each decomposed DF can be modeled by a Galois field \mathbb{F}_{2^n} , *n* is a positive integer denoting the length of binary code and 2^n is the field's size. Elements in \mathbb{F}_{2^n} are expressed as binary codes. For clarity, we assume *n* of each decomposed DF is the same. Each DF's range is then divided into 2^n discrete segments. $T \in \mathbb{F}_{2^{6n}}$ denotes a general affine transformation in 6 DFs. Acceptable margin of error can then be guaranteed on this Galois field. To quantify the error between two transformations T_1 and T_2 , following formula is defined:

$$E(T_1, T_2) = |S(I_1, I_2, T_1) - S(I_1, I_2, T_2)|.$$
(2)

Level-wise Adaptive Sampling (LAS): LAS aims to achieve a satisfactory error rate instead of testing the complete \mathbb{F}_{2^n} . From the perspective of GA, our problem can be defined as a minimization problem of SAD. In crossover operation of GA, two coded individuals swap certain portions with each other. It is a good method to span search space around a sample point in multiple directions. However, in order to optimize \hat{T} in such a broad search space, two major problems should be faced: 1) how to escape from local optimum. 2) how to control the optimization response time.

It has been argued in [2] that in order to prevent GA from falling into local optimum, genetic variety should be preserved somehow. Although mutation operation can surely increase the genetic variety randomly, it can also destroy individuals which are potentially to be close to \hat{T} . In a broad search space, the probability to create a "suitable" diversity is very low and mutation can contrarily slow down the speed of convergence. It is worth noting that in our problem, a large enough number of randomly initialized population keeps sufficient genetic variety for converging to \hat{T} . During the evolution, selection operation such as roulette wheel selection is likely to select individuals which hold larger fitness for crossover operation. With the combination of selection and crossover, genetic variety decreases and the whole population converges to an optimum solution. However, if an individual happened to hold small SAD (e.g. a candidate area is flat) in the early stage of evolution, the whole population will easily fall into a local optimum especially when the search space is very broad. To preserve genetic variety, we select individuals from each SAD level uniformly. Each SAD level is a discrete interval which is occupied by a part of individuals. With maximum SAD in m th generation defined as S_{max}^m , minimum SAD in *m* th generation defined as S_{min}^m and the number of SAD level defined as σ , we can define *i* th SAD level as $[S_{min}^m + (i-1)(S_{max}^m - S_{min}^m)/\sigma, S_{min}^m + i(S_{max}^m - S_{min}^m)/\sigma]$. Each individual which is assigned to *i* th SAD level should have a fitness within this

Graduate School of Engineering Iwate University, Japan Faculty of Engineering Iwate University, Japan



Figure 1: Heat map of matching frequency. This figure shows the frequency that each pixel has been used for calculating SAD. With the decrease of population number, the total matching frequency reduces while a more accurate candidate area can be localized.



Figure 2: Comparative results with 500 images. (a) Overlap rate error of each test image. (b) Number of matching tests of each test image.

range. Individuals of next generation are then randomly selected from each SAD level. The number of individuals selected from each SAD level is the same. With the increase of σ , distribution of SAD in m + 1 generation approximates to uniform distribution. Figure 1 shows an example of matching iteration.

Fitness uniform selection scheme (FUSS) is proposed in [2], which selects a fitness value uniformly at first and then randomly select the nearest individual. The difference is, LAS can control the degree of uniform approximation by σ , which can directly affect the convergence speed. FUSS will take a longer time to converge, because the individuals with high fitness in FUSS make up only a small percentage of overall individuals.

Experiments: We randomly generate a ground truth affine transformation matrix for each test image, and make sure that the four corers of parallelogram generated by according matrix are all in the image. Pixels in the parallelogram are then warped to generate the square template. In our experiment, each template has a size of 100×100 pixels. We compare our method [3] against the state-of-the-art method on an evaluation data set and the results are shown in Figure 2. For more detailed discussion, please read our paper. The benchmark used and our result data can be downloaded from the following URL.

http://cvhost.scv.cis.iwate-u.ac.jp/research/proj ects/affine_matching.html

- [1] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- Marcus Hutter and Shane Legg. Fitness uniform optimization. *IEEE Transactions on Evolutionary Computation (TEVC)*, 10(5):568–589, 2006.
- [3] Simon Korman, Daniel Reichman, Gilad Tsur, and Shai Avidan. Fastmatch: Fast affine template matching. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2331–2338. IEEE, 2013.