DOCTORAL THESIS

A Novel Template Matching Method based on Similarity Metric of Feature Extraction for Object Detection

Graduate School of Engineering, Iwate University
Doctoral Course, Design & Media Technology
Yi Zhang

March 2021

Contents

Chapter 1

Introduction

1.1 Summary

Template matching is a basic component in a variety of computer vision systems. It could found in various applications such as image-based rending, image compression, object detection, image matching, and action recognition, etc. The mechanism is straightforward: a large number of candidate windows are sampled in the target image, followed by a similarity measure between each pair of candidate window and template. The similarity score plays a core role in measuring the confidence of distinguishing the real target region from the candidate regions. The design of a good similarity measure is still difficult, because of the following cases. (1) The size of the target object is different in the template and target image. (2) The template includes some background regions, which differ from occlusion, noise, and appearance change. (3) The target object has some deformations in the target image (e.g., rotation, non-rigid deformation). (4) The illumination conditions differ largely between the template and the target image. (5) The target object has multiple types.

In this thesis, the feature extraction based similarity metric for object detection is discussed.

Specifically, for a single object, a universal similarity measure method is proposed that can be applied in unconstrained environments. Which is referred to as the diversity similarity measure against scaling, rotation, and illumination (DS-SRI). Specifically, DS-SRI exploits bidirectional diversity calculated from the nearest neighbor (NN) matches between two sets of points. Scaling and rotation changes are taken into consideration by introducing a normalization term on the scale change, and geometric consistency term with respect to the polar coordinate system. Moreover, to deal with the illumination change and further deformation, illumination corrected local appearance and rank information are jointly exploited during the NN search. All the features of DS-SRI are statistically assessed, and the extensive visual and quantitative results on both synthetic and real-world data show that DS-SRI can significantly outperform state-of-the-art methods for the above problems (1), (2), (3), and (4). However, DS-SRI cannot deal with the problem (5).

Furthermore, a novel rule-based similarity measure (RBSM) method is proposed. This method can measure the similarity for multiple types of objects in a complex environment, owing to that RBSM based on universal features. Similar to the DS-SRI, the template can be treated as a dictionary and utilized to check the candidate is the target or not according to some rules. Unlike DS-SRI, which confirms all pixels, the RBSM verifies some super pixels to calculate the similarity. Here, super pixels are some subregions that are grouped by the common feature of all objects.

In this thesis, two practical problems are considered to evaluate the RBSM.

The first one is the vehicle inspection sticker detection (VIS). Localization of VIS is difficult because the VIS is small and hard to be recognized due to the projective transformation and various environmental changes. Moreover, the type of VIS varies from month to month. To solve this problem, the RBSM method with adaptive background constraints is proposed that can deal with the illumination change and the projective transformation. Specifically, the matching problem is solved by treating it as an optimization problem. Which the optimization problem is solved by the genetic algorithm (GA). The experimental results show that the proposed method can robustly localize various VIS under different environments.

The second practical problem, which is handled by RBSM, is the roast fish part (RFP) detection. The various shape, sizes, and colors of the fish body parts lead us to develop an algorithm to deal with the challenge of detecting multiple RFP. To solve this problem, a RBSM based multi-object matching method is proposed. Similar to the VIS detection, the universal features of RFP are utilized to design the RBSM function and a supporting template. The RBSM function is utilized to measure the probability of the candidate being RFP. Then, a mathematical model is used to obtain the candidate regions via template mapping. Finally, GA is used to search for the local optimal solution by introducing DCAPD for multi-object detection. Our method achieves good performance with fast speed. The results of these practical problems illustrate that RMBS can cover all the above difficulties with suitable templates and rules. But this method still has some limitations. For example, the template and rules are designed manually. That will be solved in the future.

1.2 Background

Template matching, also known as pattern matching, is a vital component in a variety of computer vision applications. It is utilized to seeking a given template in a target image, as illustrated in Fig. 1.1. Template matching is widely used in computer vision, signal, image, and video processing. It can be found in varied applications such image based rending [1], quality control [2], super resolution [3], image compression [4], object detection [5], texture synthesis [6], block matching in motion estimation [7, 8], image denoising [9, 10, 11], mouth/eye tracking [12], road/path tracking [13], image matching [14] and action recognition [15]. That is surveyed by a good review [16]. Although there are many ways to match templates, no one method can be applied to all problems.

As the most crucial technique in template matching tasks, similarity measure has been studied for decades and yields in various methods from the classic methods such as the sum of absolute differences (SAD), the sum of squared distances (SSD) to recent best buddies similarity (BBS) [17], deformable diversity similarity (DDIS) [18]. Also, similarity measures have been widely applied to image processing problems such as image segmentation [19], visual tracking [20], and image registration [21].

Despite the successes of template matching, several considerable issues still need to be addressed:

- In most applications, users prefer obtaining the matching result with a free-scale bounding box to exactly including the region of the target object rather than a fixed-scale bounding box. Nevertheless, setting geometric parameters

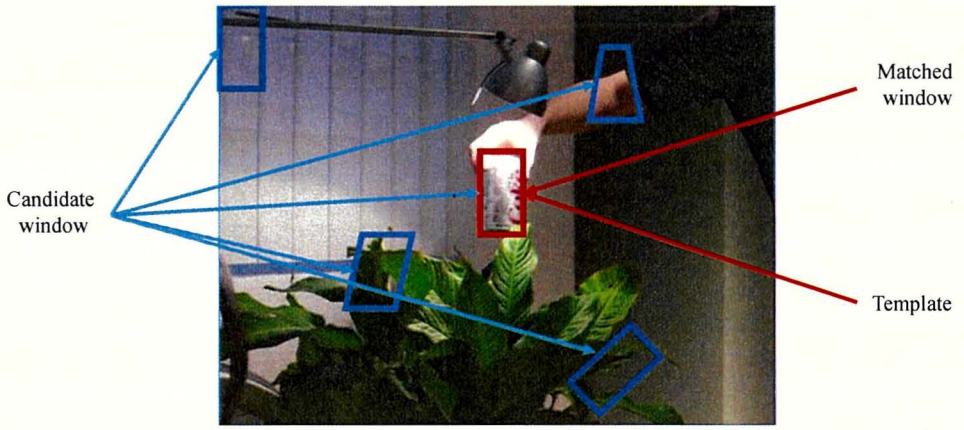


Figure 1.1: Example of template matching. The blue quadrilateral illustrates the candidate window. The red quadrilateral illustrates the template and matched result. This image is from [17].

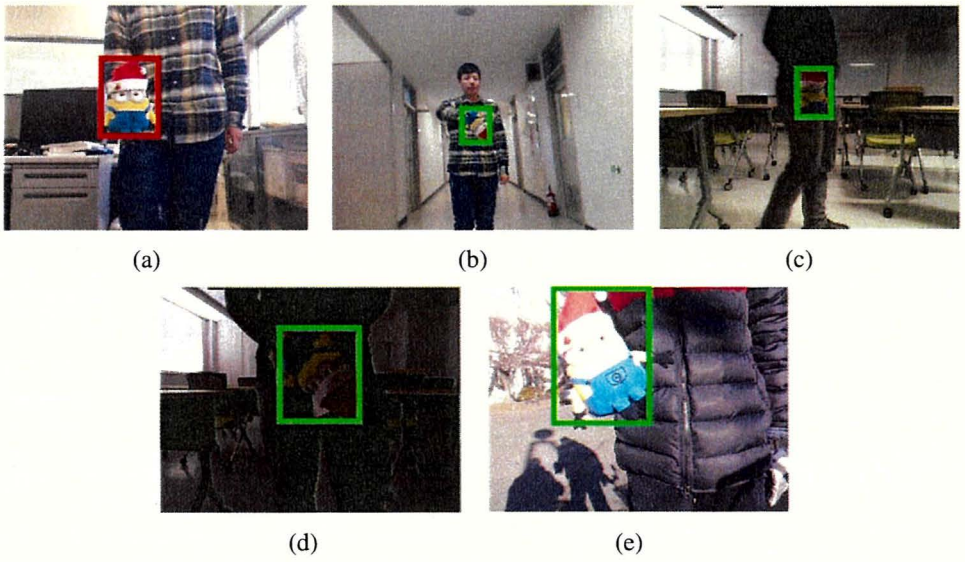


Figure 1.2: Diversity similarity against scaling, rotation and illumination (DS-SRI) for template matching. A doll is placed under different unconstrained environments. (a) Reference image. The template is marked by a red rectangle. (b), (c), (d) and (e) are the matching results over different target images with our proposed DS-SRI.

like scaling factors to the bounding box can result in an explosive growth of candidates for evaluation, forcing the similarity measure to be more discriminative to specify the matching result.

- Dense template matching usually takes all the pixels/features within the template and candidates into account to measure the similarity even some information is not desirable (e.g., occlusion, noise, appearance change), this requires a similarity measure to be consistent with noises and outliers.

- In order to deal with the deformation on the target object, a functional similarity measure is expected to be independent with the spatial correlation (e.g., when the object within a candidate window is strongly rotated, the local image patches in template and candidate are no longer spatially consistent).

- The illumination conditions can differ largely between the template and the target images, which will cause different appearances (e.g., color, intensity, contrast) and lead to a small value of similarity even between two same objects.

- Last but not least, in some scenes, the object may be not one but multiple. And all objects needed to be matched. Furthermore, the color, size, and shape of the objects are not exactly.

In this thesis, the following two scenes are focused on. One is the template matching in unconstrained scenarios. That is, a rigid/nonrigid object moves in 3D space, with variant/invariant background, and the object may undergo rigid/nonrigid deformations and partial occlusions. Besides, in this environment, the illumination

condition may be changed, as demonstrated in Fig. 1.2, that are focus on above (1), (2), (3), and (4) problems. The second is the template matching for multi-object. That is matched for a class of objects. And in this class of objects the color, size, and shape are not exactly. Which are above all problems. An example is shown in Fig. 1.3. To solve these problems two methods will be introduced as follows. The DS-SRI for the unconstrained scenarios problem and rule-based matching method for the multi-objects problem.

As to state-of-the-art methods, both BBS and DDIS are proposed mainly to settle the above issue (2) by exploiting the properties of the nearest neighbors (NNs). Here, each NN is defined by a pair of patches between the template and the target. In the case of BBS, if and only if each patch in a patch pair is the NN of the other, a match is defined and the number of such matches determines the BBS score. DDIS further improves the BBS by introducing relevant diversity of patch subsets between the target and the template, which leads to the robustness of BBS against the occlusions and deformation. Although these methods can deal with deformation within a window to some extent, there remain limitations especially on the issue **(1)**, **(3)**, **(4)**, and **(5)**.

1.3 Problem Setting

The template matching problem can be converted to a mathematical problem as follows. The target image is the input, denoted by I, with a size of $n \times m$. The purpose is to detect some objects from I. And objects are given, denoted by set o. Here, the number of the object is zero or more. The object candidates are denoted as

$c_i \in C$. And the function $S(\cdot)$ is the similarity measure function for the candidate and object. There are two different general directions to meet different practical purposes. One is detecting a specific target of o^*. An example is shown in Fig. 1.2. In this case, the target image maybe has some similar object, but our purpose is to detect the most similarity c^*. The purpose of this problem can be converted to the following:

$$c^* = \arg\max S(c, o^*) \quad c \in \mathbf{C}. \tag{1.1}$$

The second is detecting a class of objects. The class of the object maybe has some differences, noted as $o = o^* + o'$. Here, o^* respect to the part that all object is the same, o' mean the difference for each object. The purpose is to detect all objects, these objects noted as C' in the target image. An example is shown in Fig. 1.3. For detected all objects, the different part O' is ignoring. Thus, the purpose of this problem can be converted to the following:

$$C' = \delta(S(c, o*)) \quad c \in \mathbf{C}, \tag{1.2}$$

where the $\delta(\cdot)$ is a conditional function, it utilized to select the candidates based on the result of the similarity measure. When the $S(\cdot)$ meeting the rule, the c is a target candidate.

1.4 Diversity similarity measure against scaling, rotation, and illumination

In this thesis, to solve the problems (1), (2), (3), and (4), the NN pair is redefined based on the relevant diversity statistics and propose diversity similarity against scaling, rotation, and illumination changes (DS-SRI) to address all the above issues. DS-SRI can be applied with a multi-scale sliding window search for template matching, and no specific parametric deformation model is needed to be imposed on the target object.

From a general perspective, both template and each candidate can be viewed as images consist of small patches. Therefore, the visual similarity can then be viewed as a similarity measure between two point sets if treated each image patch as a point and each image as a point set. Like the first feature, DS-SRI allows similarity measure between two sets of points in different sizes, and the magnitude of the score is normalized. In contrast, the magnitude of the DDIS or BBS score grows with the increase of scales, which makes the larger candidate windows more competitive to be matched. To alleviate the unfairness caused by scaling, DS-SRI introduces bidirectional relevant diversity and normalizes scaling changes to make the employment of a multi-scale sliding window feasible. The second feature of DS-SRI is its invariance against in-plane rotation. Both BBS and DDIS involve a spatial distance term in NN search and/or the similarity calculation based on a strong prior assumption that the two points of an arbitrary NN pair from two-point sets are spatially close when plotted on the same coordinate system. This prior can indeed reduce the number of outliers of NN pairs when the object is stationary but becomes a false

constraint in the presence of large rotation. In this paper, instead of calculating spatial distance on the Cartesian coordinate, the polar coordinate is exploited to release the limitation of in-plane rotation brought by the spatial assumption. Besides, local rank information of patches is employed for searching NNs along with appearance information, which helps to find more confident NNs and yields a significant improvement when large rotation takes place.

As the last feature, DS-SRI is robust against the illumination change. NN-based methods suffer from the change of illumination because the illumination can largely affect the appearance of the target object and thus reduce the valid NN pairs between point sets. As an extreme example, if the object in the target image is exposed to intense light, all the patches of a candidate can appear white and point to the same patch in the template as the NN. In this paper, an illumination corrector is introduced to the distance function for searching the NNs. The corrector is introduced to synchronize the illumination effect on the template and the candidate. All the above features of DS-SRI are well statistically justified in Sec. 3.4.

1.5 Rule-based similarity measure

Above mentioned methods, such as best-buddies similarity (BBS) [22], deformable diversity similarity (DDIS) [18] and diversity similarity measure against scaling rotation and illumination (DS-SRI), handle the complex environment very well. Which the difficulties include non-rigid geometric deformations, background clutter, and occlusions. However, all the above-mentioned methods are only in the case of identical or high-similarity objects o^*, To use these similarity measures for a

class of objects $o = o^* + o'$, that have many differences, multiple templates may be needed, and the candidates may also need multiple measures. These requirements will increase the suffering from times cost. To solve this problem, a rule-based measure method is proposed for a class of objects.

The rule-based method can be divided into the following three-part. The first part is according to the shape and color distribution design a rule template. Then according to the common feature of targets design some rules to measure the probability. Finally, according to the rules distinguish the object from the candidate. Rule-based matching can be utilized to deal with all the above problems with the appropriate rules. However, the rule-based matching method still has some disadvantages. Which the template and rules are designed manually, and different problems need different templates and rules. In this thesis, the rule-based matching method is tested in two fact problems. Which are vehicle inspection sticker (VIS) detection and roast fish parts (RFPs) detection.

1.6 Optimization algorithm

To detect the object faster, some optimization algorithms also be introduced to reduce the candidates. For optimal candidate detection, the traditional method is a brute force, testing of all candidates, but the brute force method is difficult to use if the sample set is large. In the past decade, some more efficient optimization methods have been proposed and used in template matching, such as general GA [23] and particle swarm optimization [24], which can only efficiently acquire a global optimum. Here, the general GA is utilized to detect a single object.

There are some multi-objective evolutionary algorithms [25, 26, 27, 28] that not only acquire the global optimum but also local optima. And due to their population-based nature, evolutionary algorithms can approximate the whole Pareto set of a multi-objective optimization problem in a single running. In this thesis, a GA that uses DCAPD while distributing the population is used for acquiring the local optimal solution.

1.7 Constitution

This paper is composed of 5 chapters, the constitution is shown as follows. Chapter 1, the background of this research is introduced, and two proposed methods are simply proposed. Chapter 2, the related works are introduced from purpose and method two aspects. Chapter 3, the similarity measure method DS-SRI is introduced for unconstrained scenarios, diversity similarity measure against scaling rotation, and illumination (DS-SRI). Chapter 4, the similarity measure method for multi-object, rule-based matching method, is introduced. Chapter 5 introduce the conclusion and future work.

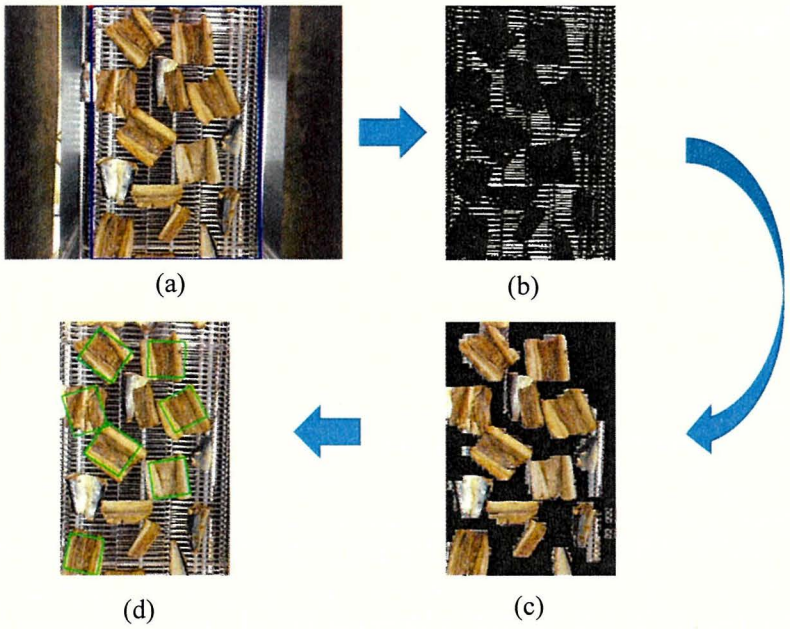


Figure 1.3: Example for image processing. (a) The target image, in which the blue rectangle marks the region of interest. (b) Wire belt segmentation result. (c) Object region estimation result. (d) Fish roast fish part direction results.

Chapter 2

Related work

In this chapter, the related work is introduced from two parts, similarity measure and Optimization algorithm.

2.1 Similarity measure

Template matching is a classic research topic mainly for object localization. The mechanism is straightforward: a large number of candidate windows are sampled in the target image, followed by a similarity measure between each candidate window and template. The similarity score plays a key role in measuring the confidence and distinguishing the target object from the background. The most widely used off-the-shelf techniques are pixel-wise methods such as SSD, SAD and normalized cross-correlation (NCC) [29, 30], owing to their simplicity and efficiency. These methods have been combined with tone mapping [31] for handling illumination change, with asymmetric correlation [32] to deal with noise.

To handle the geometric changes on the target, extending the candidate sampling

with planar parametric transformation models have been considered in many works, such as translation [32, 33, 34], similarity transformation [35, 36], affine transformation [37, 38, 39] and projective transformation [23]. However, these methods usually fail in the case of 3D deformations because the pixel-wise similarity method relies on the correct correspondences between the pixels of the template and the candidate, which is hard to be modeled by the planar transformation. Other metrics focus on improving the robustness against noise, e.g., Hamming-based distance [40, 34], M-estimators [33, 41], which are robust against the pixel-wise noise such as additive noise and salt and paper noise. The interested readers are referred to a comprehensive survey [42].

In unconstrained environments, to deal with non-rigid transformations and other noises, involving global information instead of pixel-wise local information for designing a robust similarity is a key cue. Histogram matching (HM) [43, 44, 45], which mainly measures the similarity between two color histograms, is not restricted by the geometric transformation. However, it is usually not a good choice when background clutter and occlusions appear within the candidate windows.

Another wildly used measure method is the Hausdorff distance in the context of template matching. In [46], the k^{th} farthest point to replace the traditional farthest point to deal with occlusions or degradation problem. However, k is hard to be determined in different cases. Moreover, in [47], a modified Hausdorff distance (MHD) is proposed by replacing the generalized max operator with sum to deal with noise on different levels. In [48], Hausdorff distance is used as a similarity measure between a candidate and a general face model.

Earth mover's distance (EMD) [49] is a metric for comparing sets of features,

points, and signatures that capture the distributions, and is also widely applied in template matching. It is defined as the minimum amount of work needed to change one distribution into the other distribution. EMD is robust with the deformation because it does not consider any spatial correspondence. However, EMD is difficult to deal with scaling because it requires 1:1 matching. Furthermore, a more robust approach [50] is proposed by using spatial-appearance representation to measure the EMD.

An eye-catching family of similarity measures in recent years is to explore a global statistic property over the two-point sets. Bi-directional similarity [51] proposes that two-point sets are considered similar if all points of one set are contained in the other, and vice versa. BBS [22, 17] counts the mutual two-side NNs as a similarity statistic. The DDIS [18] measures the diversity of feature matches between the two sets and is reported to outperform BBS by revealing the "deformation" of the NN field. Despite the robustness of BBS and DDIS against the transformations within the search windows, scaling and rotation on the whole search windows have not been considered. Furthermore, NN is viewed as a powerful cue in many tasks, such as image matching [52], classification of natural language data [53], image classification [54], clustering [55], etc. In this paper, the mutual nearest neighbors are exploited in the bidirectional diversity similarity.

2.2 Genetic algorithm

In template matching, another problem is the huge amount of candidates search for our targets. Genetic algorithms (GA) are introduced to improve speed. GA

[56, 57, 58, 59] are randomized sampling and optimization techniques that guided by the biology of evolution and natural genetics. GA performs a search in massive candidate space, obtain the near-optimal solution for an optimization problem. Genetic algorithms are widely used various fields, such as image processing [60], machine learning [61], neural networks [62], etc. In the area of the image process, a parameter selection method is needed to obtain optimum solutions in complex spaces. Some methods utilize the genetic algorithm to segment the image [63, 60]. Based on genetic algorithm object detection and recognition method [64, 65, 66] also is common. Furthermore, I propose a template matching method with an adaptive background model under the GA framework to localize the VIS over projective space. The proposals are also stated in [67, 68].

The traditional method is a brute force, testing of all parameters, but the brute force method is difficult to use if the sample set is large. In the past decade, some more efficient optimization methods have been proposed and used in template matching, such as GA [23] and particle swarm optimization [24], which can only efficiently acquire a global optimum. There are some multi-objective evolutionary algorithms [25, 26, 27, 28] that not only acquire the global optimum but also local optima. And due to their population-based nature, evolutionary algorithms can approximate the whole Pareto set of a multi-objective optimization problem in a single running. In this system, a GA that uses DCAPD while distributing the population is used for acquiring the local optimal solution.

Chapter 3

Diversity similarity against scaling, rotation, and illumination

3.1 Overview

In this chapter, a template matching method is introduced for a single object. The key component is behind a general similarity measure referred to as the diversity similarity measure against scaling, rotation, and illumination (DS-SRI). Specifically, DS-SRI exploits bidirectional diversity calculated from the nearest neighbor (NN) matches between two sets of points. Scaling and rotation changes are taken into consideration by introducing normalization term on the scale change, and geometric consistency term with respect to the polar coordinate system. Moreover, to deal with the illumination change and further deformation, illumination-corrected local appearance and rank information are jointly exploited during the NN search. All the features of DS-SRI are statistically assessed, and the extensive visual and quantitative results on both synthetic and real-world data show that DS-SRI can

significantly outperform state-of-the-art methods.

3.2 Nearest neighbor based similarity measure

Given a template cropped from a reference image and a target image related by unknown geometric transformation and/or photometric transformation, our purpose is to design a similarity measure, which can distinctively localize a region in the target image that exactly includes the same object of the template by maximizing the matching similarity score. Each candidate region in the target image is represented by a rectangular window, and the candidate in the target image is sampled in a way of the multiple-scale sliding window. Taking the template image $T = \{t_i\}_{i=1}^{n}$ and a candidate window $Q = \{q_j\}_{j=1}^{m}$ from target image $\mathcal{Q} = \{q_l\}_{l=1}^{M}$ as inputs, a DS-SRI score in real number can be calculated, where the t_i and q_j represent non-overlapped patch from the template and a candidate window, respectively. t_i and q_j can also be treated as points when T and Q are explained as point sets for generality. $Q \subseteq \mathcal{Q}$, and $m \leq M$.

Nearest neighbor has been shown to be a strong feature for designing similarity measure in some prior researches [17, 18]. To better address the difference, firstly BBS [17, 22] is recalled which counts the number of bidirectional NN matches between T and Q:

$$\begin{aligned} \mathrm{BBS} =& c|\{\exists t_i \in T, \exists q_j \in Q : \\ & \mathrm{NN}(t_i, Q) = q_j \wedge \mathrm{NN}(q_j, T) = t_i\}|, \end{aligned} \tag{3.1}$$

where $\mathrm{NN}(t_i, Q) = \arg\min_{q_j \in Q} \mathrm{d}(t_i, q_j)$ is a function returns the NN of t_i with

respect to Q, and the $\mathrm{d}(\cdot)$ is a distance function. $|\{\cdot\}|$ denotes the size of a set, and $c = 1/\min\{n, m\}$ is a normalization factor.

The distance function in Eq. 3.1 is defined by

$$\mathrm{d}\left(t_i, q_j\right) = \left\| t_i^{(A)} - q_j^{(A)} \right\|_2^2 + \lambda \left\| t_i^{(L)} - q_j^{(L)} \right\|_2^2, \tag{3.2}$$

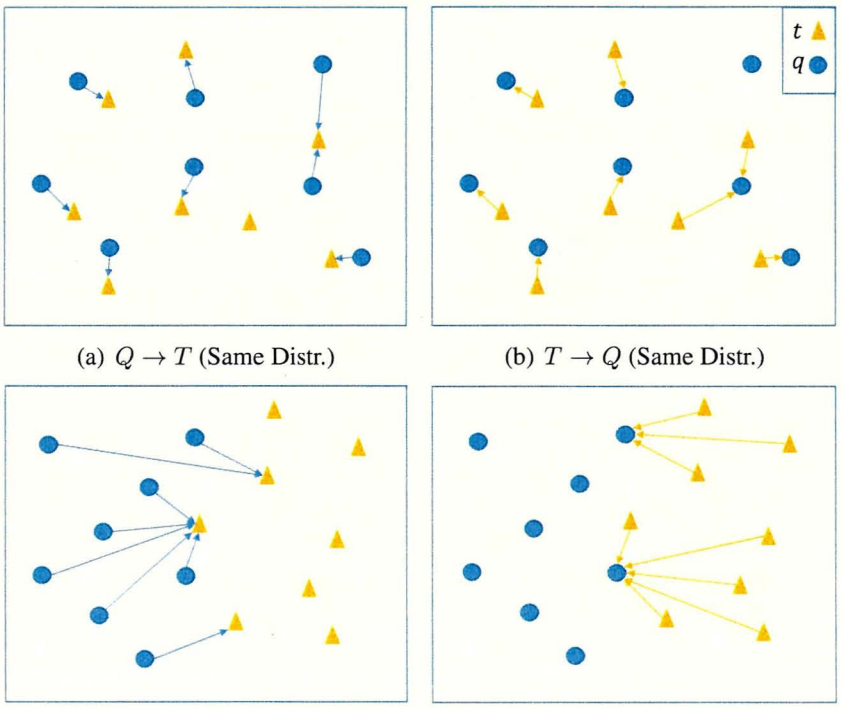
where (A) denotes pixel appearance (e.g., RGB feature) and (L) denotes pixel location (x, y) within the patch and coordinates are normalized to the range [0, 1]. In the stage of NN searching, under the assumption that illumination and large deformation do not occur within the patch, the combination of the appearance and spatial terms contribute to searching NNs by confirming the consistency of appearance and position.

On the other hand, the diversity similarity (DIS) [18] has a different usage of NNs, which is defined as

$$\mathrm{DIS} = c\left|\{t_i \in T : \exists q_j \in Q, \mathrm{NN}(q_j, T) = t_i\}\right|. \tag{3.3}$$

Where c is the normalization factor. Unlike BBS, DIS counts a certain type of point in T, which is the NN of point(s) in Q (defined as diversity in the direction of $T \rightarrow Q$).

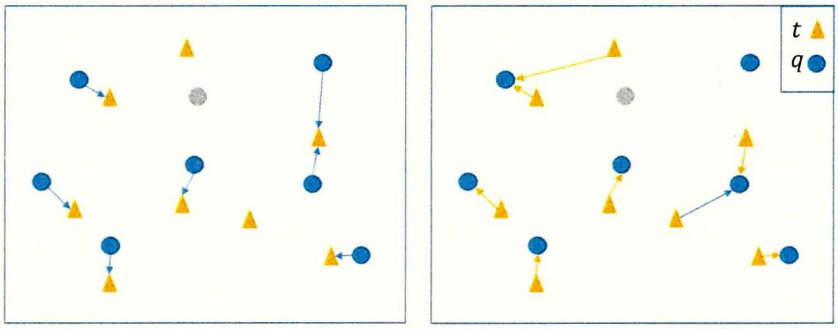
In conclusion, in order to design a good NN based similarity measure, two aspects need to be designed carefully, (1) Usage of the NNs; (2) The distance function for searching the NNs.



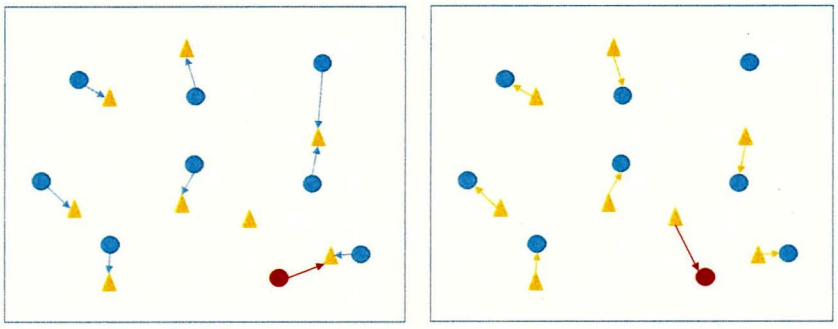
(a) $Q \rightarrow T$ (Same Distr.) (b) $T \rightarrow Q$ (Same Distr.)

(c) $Q \rightarrow T$ (Different Distr.) (d) $T \rightarrow Q$ (Different Distr.)

Figure 3.1: Intuition for distinguishing between DIS and BDS. Each arrow points to the NN of a start point. (a)(b)/(c)(d) show two examples of NN search results bidirectionally when T (circles) and Q (triangles) are drawn from the same/different distribution respectively. Different distributions can result in lower similarity. Following Eq. 3.3 ($c = 1$) and Eq. 3.6 ($\lambda_1 = 1$), DIS and BDS can be calculated from the number of end points of arrows. In (a)(b), DIS $= 7$, BDS $= 49$ (i.e., 7×7). In (c)(d), DIS $= 3$ and BDS $= 6$ (i.e., 3×2). Obviously, BDS has a greater variation in similarity value.



(a) $Q \to T$ (Downscaling) (b) $T \to Q$ (Downscaling)



(c) $Q \to T$ (Upscaling) (d) $T \to Q$ (Upscaling)

Figure 3.2: Illustration of DIS and BDS against scaling change. (a)(b)/(c)(d) show two examples of NN search results when downscaling/upscaling occurs in T (represented by circles) respectively. The gray/red circle represents for the deleted/added circle during downscaling/upscaling respectively. Following Eq. 3.3 ($c = 1$) and Eq. 3.6 ($\lambda_1 = 1$), in (a)(b), DIS $= 6$ and BDS $= 36$ (i.e., 6×6). In (c) and (d), DIS $= 7$ and BDS $= 56$ (i.e., 7×8). Obviously, the variation of BDS score is larger comparing to DIS.

3.3 Diversity Similarity against Scaling, Rotation, and Illumination

I am now ready to introduce our method in a top-down fashion: first the the DS-SRI similarity, and then the distance function for NN search.

3.3.1 Usage of the NNs

In [18], DIS is claimed as an unidirectional diversity which provides a good approximation to BBS with less computation. To design a more discriminative similarity measure, bidirectional diversity calculated is exploited with respect to T and Q (i.e., not only $T \rightarrow Q$ but also $Q \rightarrow T$). Specifically, first the following function $\varepsilon(t_i)$ is defined which indicates the number of points $q_j \in Q$ whose NNs are equal to t_i in direction $T \rightarrow Q$,

$$\varepsilon(t_i) = |\{q_j \in Q : \mathrm{NN}(q_j, T) = t_i\}|, \tag{3.4}$$

where NN($\cdot$) returns the nearest neighbor with the distance function defined in Eq. 3.8, which will be explained later.

To understand the equation, it is analyzed how Eq. 3.4 affects the diversity similarity defined in Eq. 3.3 from two situations with $|Q|$ fixed. (1) For $|T| = |Q|$, when $\varepsilon(t_i) \geq 1$, the value is inversely proportional to the diversity contribution. That is, large value of $\varepsilon(t_i)$ indicates that many points in Q have the same NN of t_i, which will lower the diversity defined in Eq. 3.3. When $\varepsilon(t_i) = 0$, it indicates that a t_i is not a NN of any q_i, which also hinders the increase of diversity similarity as no NN is utilized. It is easy to understand that an ideal situation is that for each t_i,

$\varepsilon(t_i) = 1$. (2) For $|T| \neq |Q|$, the situation becomes more complex. Assume that $s|T| = |Q|$, when $\varepsilon(t_i) = 0$, it means no contribution to the diversity similarity. Considering the scaling s between Q and T, a point in T can be the NN of multiple points in Q when $1 \leq \varepsilon(t_i) \leq s$, which will increase the value of the diversity similarity. When $\varepsilon(t_i) > s$, it will contrarily lower the maximum similarity.

The simultaneously is firstly introduced that the diversity similarity to direction $Q \rightarrow T$. This is not straightforward in the case of template matching because the candidate Q usually belongs to a target image $\mathcal{Q}$, where $|\mathcal{Q}| \gg |Q|$. That is, when finding NNs in the direction of $T \rightarrow Q$, as T is fixed and the preparation for NN search (e.g., sorting for brute force search, building kd-tree, etc.) only need to be conducted once. In the case of $Q \rightarrow T$, as such preparation for NN search has to be conducted over each Q, it will suffer from computational burden. To tackle this problem, an assumption is posed that $\mathrm{NN}(t_i, Q)$ has a high probability to be included in the set of k approximate NNs (ANNs) with respect to $\mathcal{Q}$, which is denoted by $\mathrm{ANN}^k(t_i, \mathcal{Q})$. Formally, the following function is defined which counts the number of points (i.e., patches in the image) $t_i \in T$ whose ANNs include q_j in direction $Q \rightarrow T$,

$$\tau(q_j) = \left|\left\{t_i \in T, Q \in \mathcal{Q} : q_j \in \mathrm{ANN}^k(t_i, \mathcal{Q})\right\}\right|. \tag{3.5}$$

Formally, the bidirectional diversity similarity (BDS) is proposed as following:

$$\mathrm{BDS}(T, Q, \mathcal{Q}) = \lambda_1 \sum_{q_j} \mathrm{I}(\tau(q_j) \neq 0) \times \sum_{t_i} \mathrm{I}(\varepsilon(t_i) \neq 0), \tag{3.6}$$

where $\lambda_1 = 1/(m \times n)$ is the normalization factor and $\mathrm{I}(\cdot)$ is an indicator function that turns true and false into 1 and 0. Only points in T which hold $\varepsilon(t_i) \neq 0$, and points in Q which hold $\tau(q_j) \neq 0$ can possibly contribute to the increase of the diversity.

The BDS and DIS is visually compared for clarity in Fig. 3.1 and Fig. 3.2. In Fig. 3.1, comparisons of DIS and BDS when T and Q are drawn from the same distribution (top row) and different distributions (bottom row) are illustrated. Especially, when T and Q follow different distributions, certain data points could probably become shared end points of arrows, which yields the decrease of similarity score. On the other hand, comparing to DIS, the variation of the value of BDS is larger because of the "multiplication effect", which can enlarge the gap between similar/dissimilar point sets. Furthermore, when scaling takes place, From Fig. 3.2 it can be fined that BDS score varies more largely than DIS, which can help to specify the scaling factor during matching. These characteristics of BDS will be further justified in the next section.

Based on BDS, DS-SRI is further defined to quantify the the similarity between template T and candidate Q with given target image $\mathcal{Q}$ and scaling factor $s = |\mathcal{Q}|/|T|$,

$$\text{DS-SRI}(T, Q, s, \mathcal{Q}) = \lambda_2 \frac{\text{BDS}(T, Q, \mathcal{Q}))}{\sum_{q_j} |\rho(q_j) - s\rho(NN(q_j, T))|} U. \tag{3.7}$$

Where parameter λ_2 is a normalization factor inversely proportional to the increase of s (e.g., $\lambda_2 = s^{-1}$). $\rho(\cdot)$ returns the radius of a pixel in a polor coordinate, with the pole being set at the according geometric center of T and Q. The denom-

inator of Eq. 3.7 penalizes the spatial inconsistency in polar coordinate, in order to increase the robustness against in-plan rotation. Term U is a normalization term for the number of NNs with respect to scaling. In our implementation, U is defined as $\sum_{t_i,\varepsilon(t_i)>0} \exp\left(\mathrm{I}(s/\varepsilon(t_i) \geq 1) + \mathrm{I}(s/\varepsilon(t_i) < 1)s/\varepsilon(t_i) - 1\right)$, which increases when more t_i holds $s/\varepsilon(t_i) \geq 1$. In conclusion, SR-SRI can be viewed as a similarity measure consisting of three terms: (1) The numerator term to evaluate the bidirectional diversity, (2) the denominator term to evaluate the spatial consistency, (3) the U term to normalize the number of NNs with respect to s.

3.3.2 Distance metric for NN/ANN search

Until now, the scaling-insensitivity and the rotation-insensitivity of DS-SRI are realized by BDS and polar coordinate respectively. The remaining issues are the negative effects brought by (1) illumination change, (2) large deformation, which could probably break the NN correspondence and spatial consistency by influencing Eq. 3.2. To counter the negative effects, we propose to combine the appearance and rank information for designing the distance metric for NN/ANN search. For a certain point pair of $t \in T$ and $q \in Q$,

$$\mathrm{d}\left(t,q\right) = \mathrm{disAppear}\left(t,q\right) + \lambda_3 \mathrm{disRank}\left(t,q\right). \tag{3.8}$$

Where λ_3 is a weighting coefficient. In the distance term of appearance, Gamma corrector is introduced to reduce the effect brought by illumination change, specifically,

$$\mathrm{disAppear}\left(t,q\right) = \left\| t^{(A)} - \left(q^{(A)}\right)^{1/\gamma} \right\|_2^2. \tag{3.9}$$

Where the upper script (A) means the feature of appearance, specifically the RGB color channels in our implementation. The illumination change can be caused by exposure adjustment, change of light source, appearance of shadow, etc, which can dramatically change the appearance of the target object. Gamma correction [70, 71, 72] provides a way of power law transform to equalize the imbalance between images. Here, The γ calculated from the average local gray intensities to correct each color channel,

$$\gamma = \log\left(\overline{q^{(A)}}\right) / \log\left(\overline{T^{(A)}}\right), \tag{3.10}$$

where $\overline{T^{(A)}}$ denotes the average gray intensity over the template and $\overline{q^{(A)}}$ denotes the average gray intensity over a local region in the target image. The local region can be defined as a circular (i.e., meanshift style) or rectangular (i.e., integral image style) window.

On the other hand, to deal with large deformation, we propose to utilize the rank information of local appearance, specifically,

$$\text{disRank}\,(t,q) = \left\|t^{(R)} - q^{(R)}\right\|_2^2, \tag{3.11}$$

where the upper script (R) means the rank information based on the local appearance. Take $t^{(R)}$ as an example,

$$t^{(R)} = \sum_{p \in \text{circle}(t,r)} \mathrm{I}\left(q^{(A)} \geq t^{(A)}\right) / r^2. \tag{3.12}$$

In the case of 3-channel q and t, the indicator function counts the number of chan-

nels in which the value of q is greater. The origin of the circle window $\text{circle}(t, r)$ is the coordinate of t, with a support radius of r. r^2 is a normalization term. The appearance rank defined by Eq. 3.12 is insensitive to local geometric changes, which can also be considered as structural information (e.g., the shape of the distribution of pixel values) extracted from a local region. As geometric changes can hardly destroy this structure, it is reasonable to explain its insensitivity against rotation and certain deformations.

3.4 Statistical Analysis

In this section, the features of DS-SRI described is statistically analyzed in the previous section for justification, including scaling-insensitivity, rotation-insensitivity, and illumination-insensitivity of the matching results.

3.4.1 Analysis of scaling-insensitivity

One important feature for a robust metric is the ability to preserve the similarity score of the same object against scaling change. To assert this feature in DS-SRI, in Fig. 3.3, a 1D statistical analysis is first provided as following [22, 18]. The expectations of similarity between two point sets drawn from two different 1D Gaussian models are calculated for comparison, where point sets are cast as template/candidate, points are cast as patches. Monte-Carlo integration is utilized for approximating the expectation as suggested in [18]. The first observation from Fig. 3.3 is that the expectation of DS-SRI is maximal when the two Gaussian models are the same and decrease fast when models separate. The second observation from

Fig. 3.3 is that DS-SRI is scaling-insensitive, i.e., the heat maps of Fig. 3.3(g), (h), (i) are almost the same.

Another important feature to confirm is whether the scaling factor of the target object with respect to T can be appropriately estimated by maximizing DS-SRI.A statistical result is provided in Fig. 3.4. Similar with Fig. 3.3, T is drawn from $N(0,1)$ and Q is generated from another source for the generation of expectation map. The difference is, I further prepare $\mathcal{Q}$ which involves not only Q but also background points to simulate the template matching task. Here, $\mathcal{Q} = T \cup B$, $GT_s|T| + |B| = |\mathcal{Q}|$ and B is composed of background points drawn from $N(\mu, \sigma)$, with $\mu \in [0, 10], \sigma \in [0, 10]$. In this demonstration, $|T|$ and $|\mathcal{Q}|$ are set to 100 and 200 respectively. $|Q| = s|T|$ and s varies from 0.5 to 2 with step of 0.1. The Q can be treated as a candidate window in the template matching task and is sampled from $\mathcal{Q}$ by preferentially sample points in T (i.e., nearest neighbor interpolation). For example, when $s = 1.5$, 150 points need to be sampled to construct Q, with 100 points from T and 50 points from B. Estimated $\hat{s} = \arg\max_s \text{DS-SRI}(\text{T}, \text{Q}, \text{s}, \mathcal{Q})$ is supposed to approximate the ground truth scale GT_s well. In Fig. 3.3(a), we can observe that high expectation values of DS-SRI distribute more densely around the diagonal comparing other methods. These statistical analyses clearly show the robustness of DS-SRI against scaling change, and the ability for estimating the proper scale of the target object.

3.4.2 Analysis of rotation-insensitivity

To show the robustness against rotation, the expectation of similarity is analyzed between two sets T and Q drawn from 2D Gaussian models. As shown in Fig. 3.5,

the parameters are fixed except θ and σ_2 to validate the effect of rotation angle with the Gaussian of fixed shape. As points in T and Q are exactly the same except the rotation angle, the similarity between T and Q is expected to be the maximum value no matter how the θ varies. In the case of BBS, as we can observe from Fig. 3.5 (c) when σ_2 is extremely small, the points drawn are likely to form a line, which is sensitive to rotation as the intersection of two lines is small. This is also the case when $\sigma_2 \gg \sigma_1$, as it can be observed that the expectation decreases gradually with the increase of σ_2. Also, isotropic Gaussian is supposed to be unaffected by the rotation, which can be convinced from Fig. 3.5 (c) that when $\sigma_1 = \sigma_2 = 1$, the expectation keeps well with respect to the rotation. On the other hand, SDS and DS-SRI show the invariance to the rotation despite the shape change of distribution in Fig. 3.5 (d) and Fig. 3.5 (e) .

3.4.3 Analysis of illumination-insensitivity

To show the robustness against illumination, the expectation of similarity is analyzed between two sets T and Q that drawn from 1D Gaussian models $N(0,1)$ and $N(0,\sigma), \sigma \in [0,10]$ respectively. Moreover, to simulate the illumination change, the Q is Gamma corrected with random $\gamma \in [0.5,2]$. The results are shown in Fig. 3.6. It can be observed that the expectation of DS-SRI is almost constant and approximates to 1. However, other metrics including BBS, DDIS, and our previously proposed SDS decreases gradually when the γ gets away from 1. Also, the value of DS-SRI drops fastest when σ gets away from 1. Similarly, this can be observed in Fig 3.7 when the σ and verify μ is fixed from 0 to 10 for sampling Q. We can also observe that only DS-SRI shows a high expectation value around the setting of the

Table 3.1: Average overlap rate. The red and blue colors indicate the best and the second best method respectively.

Data category	#Images	DS-SRI*	DS-SRI	SDS*	SDS [69]
Scaling-change-only	166	0.66	0.45	0.67	0.45
Rotation-change-only	166	0.58	0.60	0.59	0.60
Illumination-change-only	112	0.63	0.64	0.28	0.30
All-change	280	0.54	0.42	0.49	0.40
ALL	724	0.59	0.50	0.52	0.44
Data category	DDIS*	DDIS [18]	BBS [17]	HOG [73]	HM
Scaling-change-only	0.43	0.44	0.38	0.28	0.38
Rotation-change-only	0.40	0.53	0.43	0.18	0.36
Illumination-change-only	0.24	0.39	0.37	0.55	0.15
All-change	0.31	0.38	0.35	0.13	0.22
All	0.35	0.43	0.38	0.24	0.26

template ($\mu = 0$) with respect to the change of γ.

3.5 Experiment of DS-SRI

To show the statistically justified features of DS-SRI can really help to improve the performance of template matching task on real-world data, a comprehensive experiment is conducted with both qualitative and quantitative tests to validate the superiority of DS-SRI comparing with the state-of-the-art methods BBS [22, 17], DDIS [18], our previous work SDS [69], as well as several conventional methods [18, 17, 73].

3.5.1 Dataset

For comparison, 724 image pairs (reference-target pair) are originally collected under different unconstrained environments and categorize to create a dataset for eval-

uating the performance of template matching involving overall scaling, rotation, and illumination changes on the target object. Besides, these images also include other uncontrolled challenges like complex deformations, occlusion, background clutter, etc. The bounding box of each ground truth is annotated manually image by the image with a free-scale rectangle. The dataset is further subdivided into four categories: (1) *scaling-change-only*, (2) *rotation-change-only* (3) *illumination-change-only* and (4) *all-change* for detail evaluation and discussion, which include 166, 166, 112, 280 reference-target image pairs, respectively. It is noteworthy that each category also includes other uncontrolled photometric and geometric transformations as they are taken under unconstrained environments.

3.5.2 Quantitative Evaluation

The same procedure is followed as suggested in [22, 18] for a fair comparison. As to the evaluation criterion, following [22, 18], The success ratio is employed based on the overlap rate between ground truth W_g and matching result W_r to measure the accuracy, which is defined as: $|W_r \cap W_g| \,/\, |W_r \cup W_g|$. Here, the operator $|\cdot|$ is to count the number of pixels within a window. In the template matching task, similarity metrics have to be combined with search methods. For clearness, when both single-scale and multi-scale search methods are compared for the same similarity metric, the $\{\cdot\}^*$ is used to denote the approach that combined with a multi-scale search window. The search method is fixed to the sliding window in the experiment. Note that only DS-SRI and SDS are originally designed to be employed with search windows in multi-scale. For fairness, as a reference, the performance of DS-SRI is simultaneously compared with a single-scale search window. In ad-

dition, the DDIS also employed with a multi-scale search window for comparison, denoted as DDIS*.

We compare our proposed methods (DS-SRI, DS-SRI*) to DDIS, DDIS*, SDS, SDS*, BBS, HM, HOG, and SAD. In the case of a fixed-scale search window, the window size of candidates equals the size of the template. In the case of a multi-scale search window, the scaling factor with respect to both x and y axes range from 0.5 to 2, with step 0.1. The patch size of DS-SRI, SDS, DDIS, and BBS patch is fixed to 2×2 pixel. The results are reported in Fig. 3.8. As we can observe from Fig. 3.8(a) and Fig. 3.8(b), the performance of DS-SRI/DS-SRI* against scaling and rotation changes are almost the same as our previous work SDS/SDS* with respect to the area-under-curve (AUC) score. In Fig. 3.8(a), multi-scale approaches DS-SRI* and SDS* largely outperform their fixed-scale versions DS-SRI and SDS. On the other hand, in Fig. 3.8(b) and Fig. 3.8(c), since no large scaling changes are involved, multi-scale approaches did not show the advantage. In Fig. 3.8(c), the proposed DS-SRI/DS-SRI* shows its superiority over-illumination change against other methods. Also, in Fig. 3.8(d), which involves all the changes, our method outperforms others. Fig. 3.9 averages the success curves over all the data (Fig. 3.8 (a)$\sim$(d)) to summarize. Also, the HOG feature is compared to assess the reasonableness of each data category. Rather than a similarity metric, HOG is a gradient-based feature descriptor calculated from a uniformly spaced dense grid of blocks and cells and is known to be robust against illumination change and weak on deformations. As expected, HOG based matching performs well against illumination change in Fig. 3.8(c), but is ineffective to deal with scaling and rotation in Fig. 3.8(a)(b). The comparison of average overlap rate between results and ground truths are summa-

rized in TABLE 3.1, proposed DS-SRI/DS-SRI* achieves the best result over three categories out of four, results in the overall best performance.

3.5.3 Qualitative Evaluation

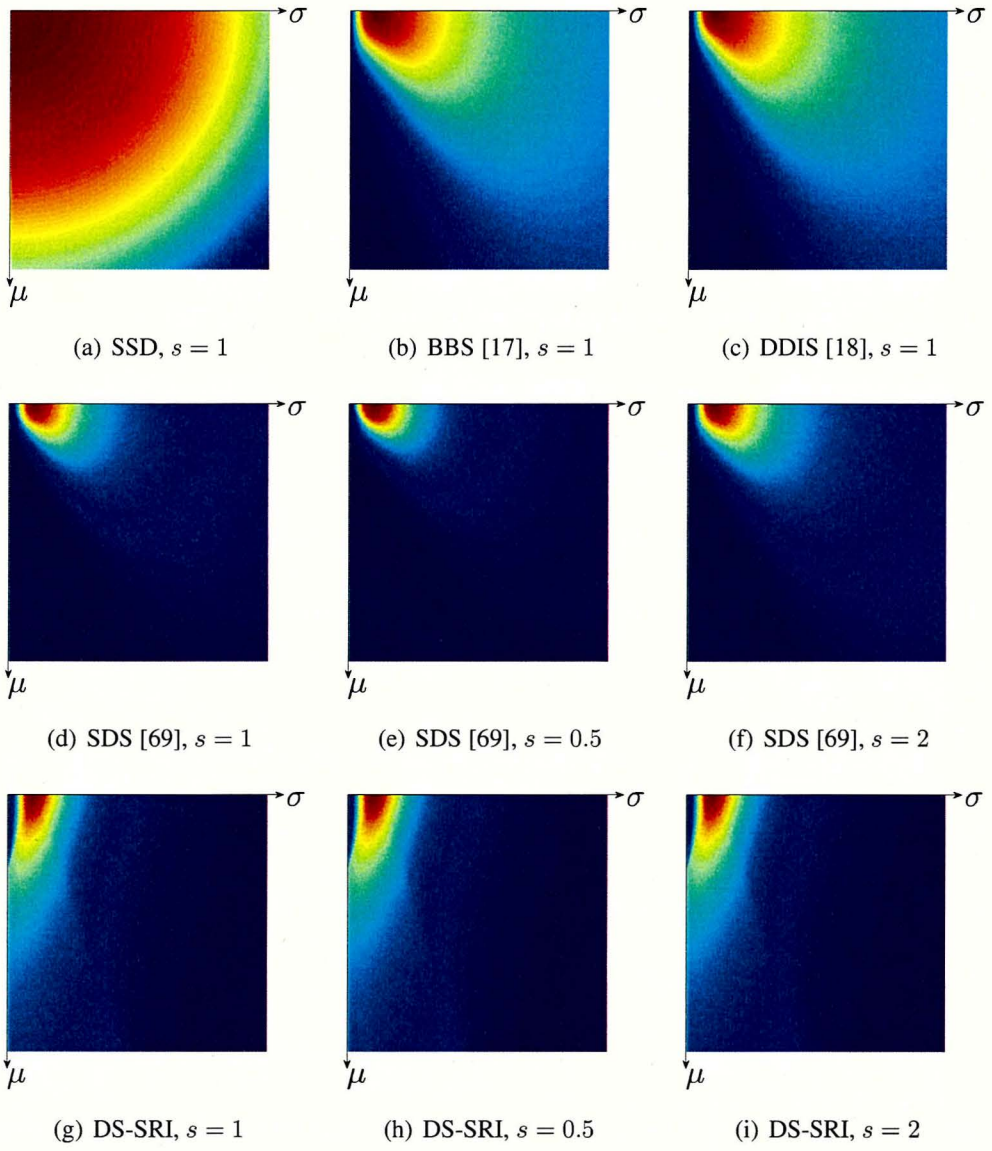
Matching examples are shown in Fig. 3.10. 1st~2nd rows, 3rd~4th rows, 5th~6th rows are the example results from category *scaling-change-only*, *rotation-change-only*, and *illumination-change-only* respectively. Example results from *all-change* are shown in the last three rows. As we can observe, the proposed DS-SRI/DS-SRI* is the only method correctly matching the template in all the challenging examples. By observing Fig. 3.10(b) and Fig. 3.10(c) we can find that the likelihood maps of DS-SRI* and DS-SRI are almost the same, which is evidence to indicate that our method is robust against scaling change. Furthermore, comparing to the state-of-the-art method DDIS (Fig. 3.10(e)), we can clearly find that our method (Fig. 3.10(c)) largely outperforms since high similarity values are mostly calculated on the object of interest and drop faster when the candidates getaway. In general, the likelihood maps of DS-SRI/DS-NSRI are more distinct and yield in better-localized modes. In the 6th row, compared to the face in the reference image, there is a very large change in illumination and facing direction. DDIS is trapped by a similar pattern in the background while our method can distinguish the face from the background clutter.

3.6 Conclusion

A novel multi-scale template matching method is proposed in unconstrained environments, which is robust against scaling, rotation, and illumination changes. Also, it takes advantage of the global statistic to deal with complex deformations, occlusions, etc. Extended bidirectional diversity combined with rank-based nearest neighbor search forms a scale-robust similarity measure, and the exploit of polar coordinate further improves the robustness against rotation. Moreover, in order to deal with the illumination change and further deformation, illumination-corrected local appearance and rank information are jointly exploited during the NN search. The experimental results have shown that DS-SRI can remarkably outperform other competitive methods.

Despite the robustness of our method, it still has a few limitations. It is likely to mislocate the object when the color distribution of the template is flat. It is also the case when the patches in the template are similar to each other. And it can not deal with the multiple object case.

In future work, I would like to develop effective scale search methods to reduce the number of similarity calculations and thereby the computational cost. I would also like to apply DS-SRI with high-level features like deep features to improve the matching performance.

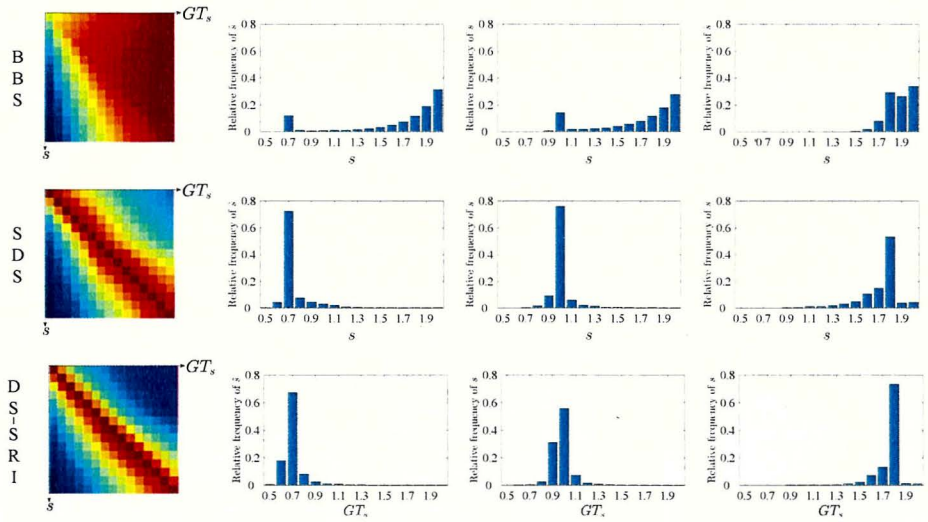


(a) SSD, $s = 1$ (b) BBS [17], $s = 1$ (c) DDIS [18], $s = 1$

(d) SDS [69], $s = 1$ (e) SDS [69], $s = 0.5$ (f) SDS [69], $s = 2$

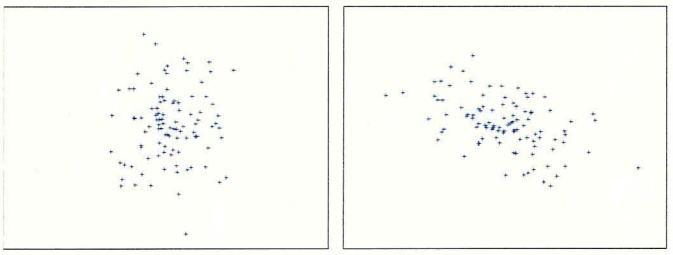
(g) DS-SRI, $s = 1$ (h) DS-SRI, $s = 0.5$ (i) DS-SRI, $s = 2$

Figure 3.3: Expectation maps of SSD, BBS [17], DDIS [18], SDS [69] and DS-SRI in 1D Gaussian case. Two points sets, T and Q are randomly drawn from two 1D Gaussian models $N(0, 1)$ and $N(\mu, \sigma)$, respectively. $\mathcal{Q}$ is set to be the same with Q. All of point are normalized within [0,1]. In (a)$\sim$(d), (g), $|T|$ and $|Q|$ are set to 100 (i.e., with a fixed scale). In (e)(h), $|T| = 100$ and $|Q| = 50$ (i.e., $s = 0.5$). In (f)(i), $|T| = 100$ and $|Q| = 200$ (i.e., $s = 2$). In each graph, the parameters of the Gaussian for generating Q increase from left-top ($\mu = 0, \sigma = 0$) to right-bottom. It can be clearly observed that SDS and DS-SRI drop faster than other methods when ($\mu \neq 0, \sigma \neq 1$), and DS-SRI preserve the map best against scale change.

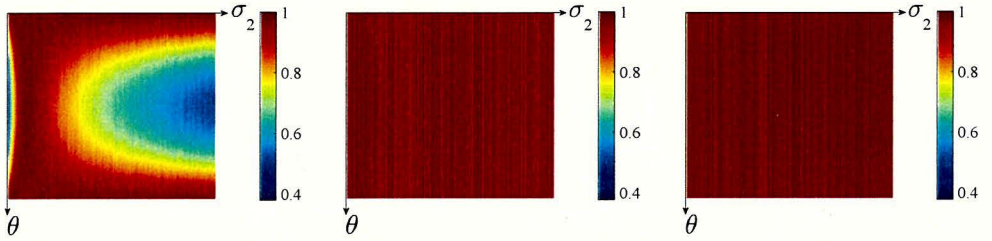


(a) Expectation (b) Ground truth s=0.7 (c) Ground truth s=1.0 (d) Ground truth s=1.8

Figure 3.4: Scale estimation by similarity maximization. (a) shows the expectation map concerning the variation of ground truth GT_s and estimated s. SDS and DS-SRI (second row and bottom row) achieve maximum expectation values on the diagonal while BBS (top row) fails in estimating the proper scale. High expectation values of DS-SRI distribute more densely around the diagonal comparing other methods. (b)$\sim$(d) demonstrates the normalized histogram of estimated s based on 200 random trials. In the case of SDS and DS-SRI, the according bin of $s = GT_s$ achieves the highest frequency. BBS (top row) performs well in a local scale range while fails in the global.

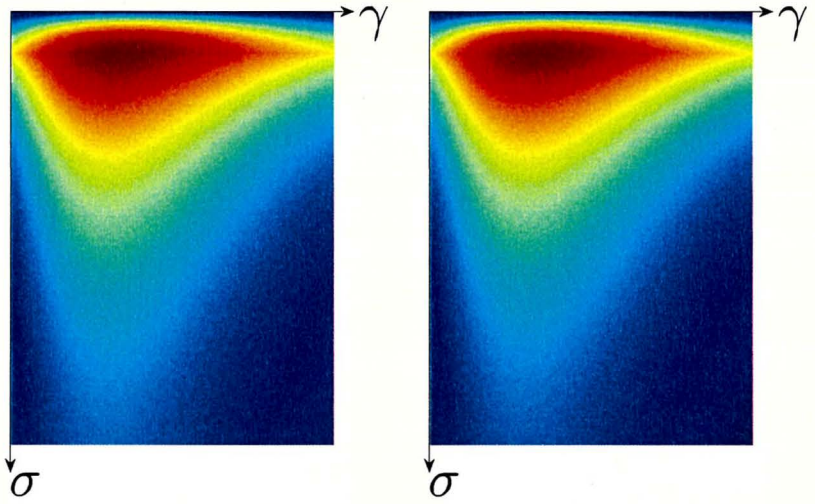


(a) Example of T (b) Example of Q

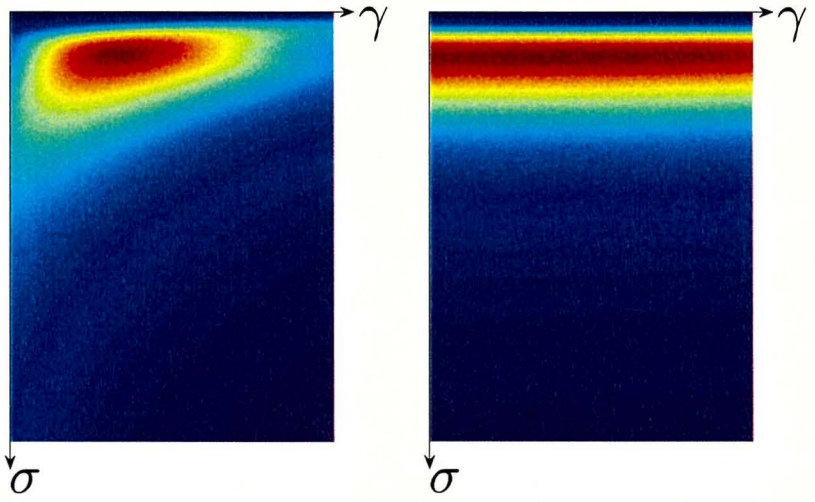


(c) BBS (d) SDS (e) DS-SRI

Figure 3.5: The expectation maps of BBS, DS-SR, and DS-SRI in 2D Gaussian case with rotation. Points in T are drawn from $N(\mu, \sigma_1, \sigma_2)$, with $\mu = (0, 0)$, $\sigma_1 = 1$, and $\sigma_2 \in (0, 10]$. Points in Q are copied from T and further rotated by θ, $\theta \in [0, -\pi]$. (a) shows an example of T and (b) is generated by rotating (a). (c), (d) and (e) are the expectation maps of BBS, SDS and DS-SRI respectively by varying θ and σ_2. It can be clearly observed that the expectation of SDS and DS-SRI is almost invariant to rotation while BBS drops most when T and Q overlap least (i.e., $\theta = -\pi/2$).

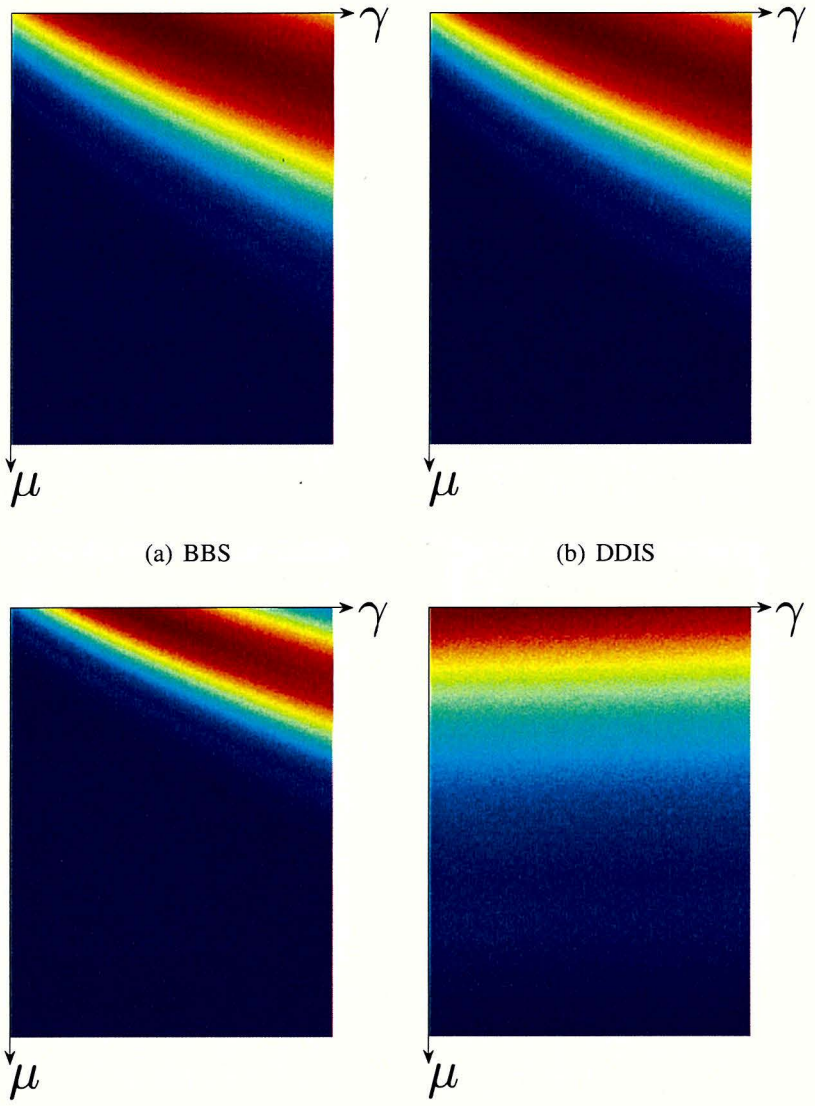


(a) BBS (b) DDIS



(c) SDS (d) DS-SRI

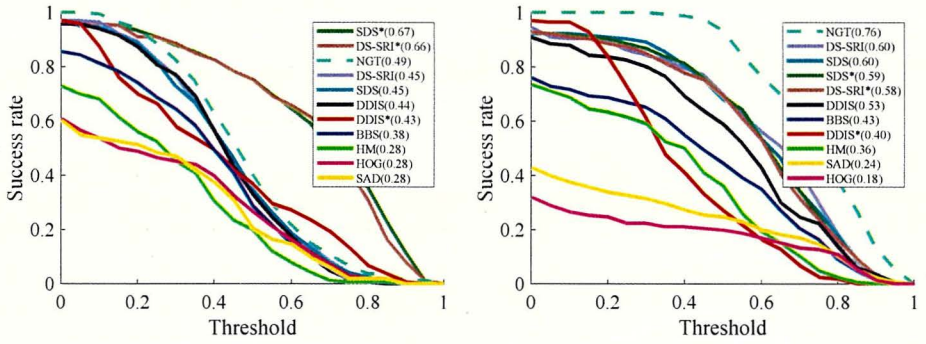
Figure 3.6: Expectation maps of BBS, DDIS, SDS and DS-SRI in 1D Gaussian case with illumination change (simulated by gamma correction). Two points sets, T and Q are randomly drawn from two 1D Gaussian models $N(0,1)$ and $N(0,\sigma), \sigma \in [0,10]$, respectively. $\mathcal{Q}$ is set to be the same with Q. In (a), (b), (c) and (d), $|T|$ and $|Q|$ are set to 100 (i.e., fixed scale). The parameters for generating Q increase from left-top ($\gamma = 0.5, \sigma = 0$) to right-bottom, $\gamma \in [0.5, 2]$ and $\sigma \in (0, 10]$. It can be clearly observed that the expectation of DS-SRI value is invariant against γ increase and drops shapely when σ gets away from 1.



(a) BBS (b) DDIS

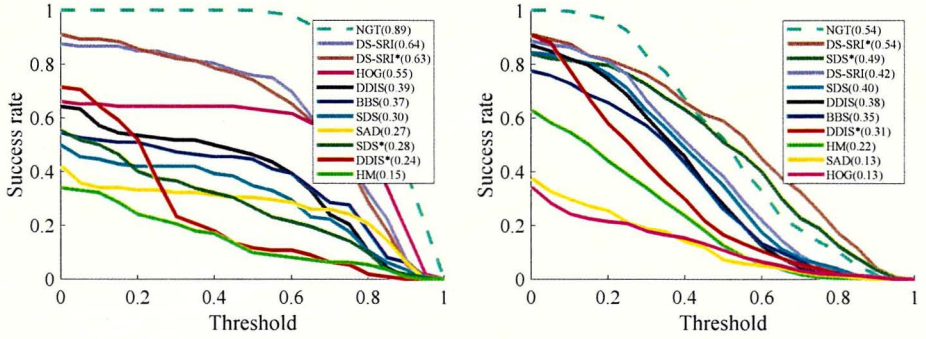
(c) SDS (d) DS-SRI

Figure 3.7: Expectation maps of BBS, DDIS, SDS and DS-SRI in 1D Gaussian case with illumination change (simulated by gamma correction). Different to Fig. 3.6, T and Q are randomly drawn from two 1D Gaussian models $N(0,1)$ and $N(\mu,1), \mu \in [0,10]$, respectively.



(a) Success curves over *scaling-change-only* category

(b) Success curves over *rotation-change-only* category



(c) Success curves over *illumination-change-only* category

(d) Success curves over *all-change* category

Figure 3.8: Comparison on success rate with respect to the variation of the overlap rate threshold. DS-SRI* , SDS*, and SDDIS* run with multi-scale search window and others are with fixed-scale. The dotted curve NGT is the performance of the ground truth with fixed-scale search window for reference (i.e., each ground truth of NGT is represented by a bounding box which has the centroid of the annotated ground truth and the size of the template). Numbers in the legend are the AUC values, i.e., the average success rate with respect to each curve. (a), (b), (c) and (d) show the success curves over four categories (*scaling-change-only*, *rotation-change-only*, *illumination-change-only*, *all-change* respectively. Best viewed in color.

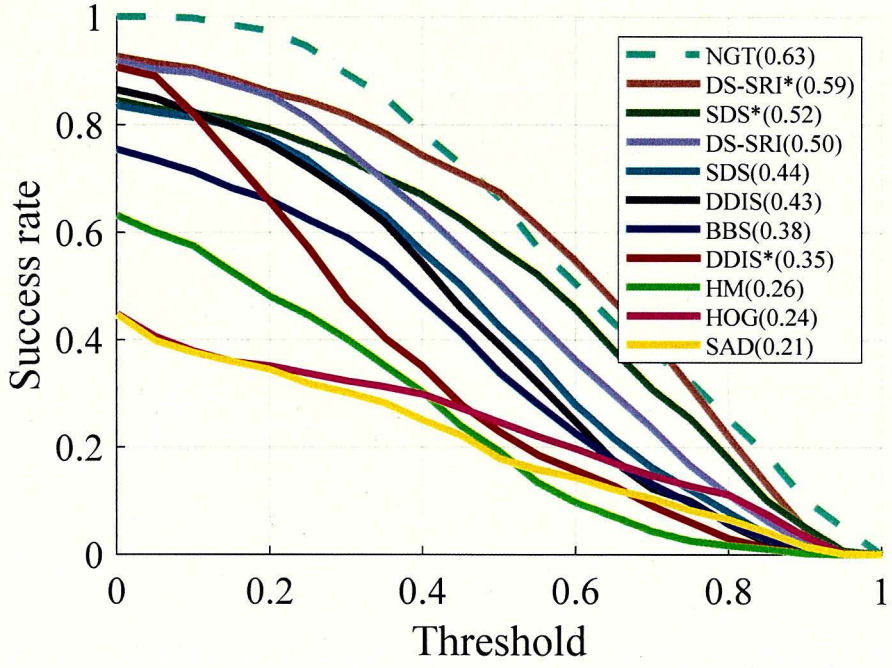


Figure 3.9: Comparison on success rate over all the data (Fig. 3.8 (a)∼(d)).

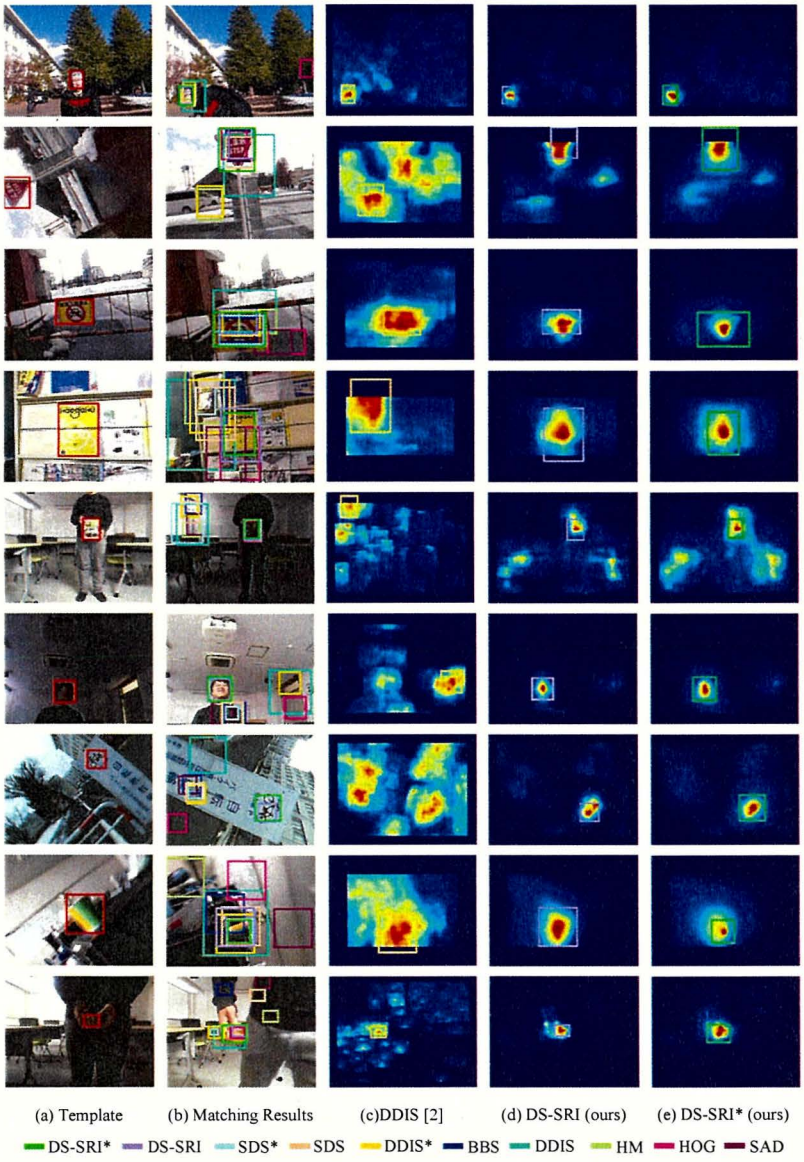


Figure 3.10: Examples of matching results. (a) The template is represented by a red rectangle. (b) The plot of detected bounding boxes. (c)∼(e) The likelihood maps of DS-SRI*, DS-SRI and DDIS, respectively. The candidate window with the global maximum similarity in each map is selected as the final matching result. In the likelihood map of DS-SRI*, every pixel has multiple similarity values due to multi-scale candidates, and only the maximum one is shown.

Chapter 4

Rule-based similarity measure

Although DS-SRI deals with the single object problem very well, it cannot utilize for a class object $o^* + o'$ detection. To solve the disadvantage of DS-SRI, the rule-based similarity measure is proposed. In this chapter, the rule-based similarity measure method will be introduced from the following aspects. (1) designing a template for a class of objects. (2) determining candidate by mathematics model and template. (3) making rules for measure candidates. (4) searching object from all candidates.

4.1 RBSM

4.1.1 Template

In common template matching, the template is selected from a reference image, that the user wants to detected or tracked. However, in rule-based matching, it is different. The template is designed manually according to the universal feature distribution of the objects. In this thesis, only the RGB feature is discussed, but it

also can be utilized in other features easily. Firstly, the shape of objects is utilized to design the template shape. Then, according to the object color distribution, the template is divided into several areas. The template needs to descry the objects common part O^*. The template is noted as T.

4.1.2 Candidates

The candidates are decided by the practical problem, it needs to cover all possible. And all candidates C are obtained by template T with a mathematical model M. The mathematical model will decide the number of the candidate. A good template can reduce the complexity of the mathematical model. That can reduce the number of the candidate.

4.1.3 Rules

The rules are designed according to the color distribution. Rules are utilized to measure that the candidate is the object or not. For detected the object, these rules need to measure the similarity of the common part O^*, and ignore the different part O'. And the rules come with the template. In this thesis, the pixel relationship between different regions of the template is considered as a feature. These rules are represented by the function R.

4.1.4 Optimization

With the rules and candidates is defined, the object can be selected from the whole candidate set. However, when the mathematics modal, that is utilized to gener-

ate the candidate combines with the template, is complex, the candidate number will be a huge amount. The brute force search is an exhaustive search for all candidates is very time-consuming, due to the massive number of candidates. Thus some optimization algorithms are needed to reduce the candidates. And due to their population-based nature, evolutionary algorithms are able to approximate the whole Pareto set of a single-objective/multi-objective optimization problem in a single running. In the above section, the candidates and similarity measure function is defined. The DS-SRI can be detected the object very fast by exhaustive search, results from the candidate are not many candidates. However, for some cases of RBSM, the rotation and deformable cover by the candidates. That is exhaustive search for all candidates is very time-consuming, due to the massive number of candidates. Thus some optimization algorithms are needed to reduce the candidates. And due to their population-based nature, evolutionary algorithms can approximate the whole Pareto set of a single-objective/multi-objective optimization problem in a single running.

In this thesis, the optimization algorithm under the genetic algorithm framework. GA is a search heuristic that is inspired by Charles Darwin's theory of natural evolution. This algorithm reflects the process of natural selection where the fittest individuals are selected for reproduction in order to produce offspring of the next generation. In GA, a very important notion is natural selection. The process of natural selection starts with the selection of the fittest individuals from a population. They produce offspring which inherit the characteristics of the parents and will be added to the next generation. If parents have better fitness, their offspring will be better than parents and have a better chance of surviving. This process keeps on iterating and in the end, a generation with the fittest individuals will be found. This

notion can be applied to a search problem. We consider a set of solutions for a problem and select the set of best ones out of them.

Five phases are considered in a genetic algorithm. 1). Initial population. 2). Fitness function. 3). Selection. 4). Crossover. 5). Mutation. In the following, these phases will be introduced one by one.

Initial population

In the GA, the process starts with a group of individuals, which is called a population. Each individual is the solution to the problem we are trying to solve. An individual is characterized by a set of parameters (variables) called genes. The genes are linked together into a string that forms a chromosome (solution). The set of genes of an individual is represented as a string. Usually, binary values (strings of 1 and 0) are used. The genes are encoded in a chromosome. In the template matching, each individual is a candidate region. And the parameters of the candidate are coded by chromosome.

Fitness function

Fitness function determines an individual's level of fitness (an individual's ability to compete with other individuals). It gives each individual a fitness score. An individual's probability of being selected for breeding is based on its fitness score. In template matching, the fitness function is the similarity measure function designed by the user.

Selection

The idea behind the selection phase is to select the most suitable individuals to pass on their genes to the next generation. Two pairs of individuals (parents) are selected based on their suitability scores. Individuals with high fitness scores have a greater chance of being selected for reproduction. In our problem, the suitable individual means the high singularity score of the candidate.

Crossover

Crossover is the most important stage in the genetic algorithm. For each pair of parents to be mated, an intersection is randomly selected from the genes. There is various crossover method in the existing literature. Such as single-point crossover, two-point, and k-point crossover, uniform crossover, crossover for ordered lists.

Mutation

In certain new offspring formed, some of their genes can be subjected to a mutation with a low random probability. This implies that some of the bits in the bit string can be flipped. The mutation occurs to maintain diversity within the population and prevent premature convergence.

Termination

If the population has converged (does not produce offspring that are significantly different from the previous generation), then the algorithm is terminated. Then it can be said that the genetic algorithm has provided a set of solutions to our problem.

4.2 Rule-based matching for VIS detection

Two practical examples are considered to introduce the rule-based matching method. These examples are the vehicle inspection sticker (VIS) and roast fish parts (RFP) detection. In these examples, the target object is a class of objects. In VIS detection, there is only one object in a target image. But for the RFP detection, there are maybe multi-object in a target image. For these problems, the background, designed template, candidates, rules, as well as optimization will be introduced.

4.2.1 Background

Image processing technology has been widely applied in vehicle-related researches. However, vehicle inspection sticker (VIS) detection and recognition have not been widely studied. Inspecting whether a vehicle inspection is expired or not still depends on the manual check. The high cost of human labor leads to the fact that only a small portion of VIS can be inspected. As a result, some drivers will keep driving with expired vehicle inspection because of the high cost of vehicle inspection or other reasons, which is a great security risk.

On the other hand, the vehicle inspection sticker (VIS) is issued by the specialized agencies after the annual inspection is qualified, the expired date is written on the vehicle inspection. In order to show the public and the traffic police that the vehicle inspection has not expired, the relevant laws and regulations stipulate that the VIS must stickered on the front window of the vehicle. Therefore, I can obtain the vehicle inspection expired date from the image of the front window. It will be very convenient if VIS can be automatically detected by a single camera on the image of

the front window.

However, it is difficult to localize the VIS from a single image, which is affected by the following factors. 1) The VIS is very small compared with the license-plate. 2) There exists a complex position relationship between VIS and camera, such that VIS is usually perspectively transformed in the target image. 3) The VIS changes appearance under different illumination conditions. 4) The feature of VIS is difficult to utilize for simplicity, this owing to there are different 12 characters on the VIS. Thus, applying local features to the localization of the VIS is difficult.

4.2.2 Problem description

Target image in grayscale is the input, denoted by I_1, with size of $n_1 \times m_1$. Normalizing each pixel value of I_1 from [0,255] to [0,1]. According to the pixel value distribution of VIS, I create a template with a size of $n_2 \times m_2$ pixel, denoted by M. Candidate region I_c is M mapped to I_1 by homographic $\delta \in \mathbf{PS}$, while δ is the projectivity operation matrix, shown in Fig. 4.1. An arbitrary pixel in M is denoted by p, while p^δ corresponds to a pixel of I_c. Based on the pixel value distribution of VIS, I create some rules to measure the similarity between VIS and I_c. According to these rules, the rule-based similarity function F is defined, then F is utilized to measure the fitness of I_c. And the score of F is smaller, the degree of similarity is higher between the VIS and I_c. With the F defined, the VIS localization can be converted to the problem that detection optimal δ^* to minimize F in projection space $\mathbf{PS}$.

$$\delta^* = arg\ min\ F(I_1, M, \delta) \quad \delta \in \mathbf{PS}. \tag{4.1}$$

4.2.3 Without character information template

In this method, The optimum region is searched in the projective space, thus the template size does not directly impact our result. In order to calculate speed, the template size is set as small,32 $\times$ 32 pixels, denoted by M_2, shown in 4.2. Further, I divided M_2 into two parts, including the VIS area and the front glass area, respectively denoted by M_2^v and M_2^o. The position of M_2^v is in the center of M_2, with the size is 16 $\times$ 16 pixels. The template is shown in Fig. 4.2.

4.2.4 Candidates

In this section, I detailed define projective transformation δ. According to the pin−hole camera model we can know that the 2D projective transformation can be viewed as a transformation within the 3D and then projected onto a 2D plane. It is comprising by eight simple transformations on 3D, shown in 4.3. These transformations include scaling with X and Y axes, rotation with X, Y and Z axes, translation with X and Y axes. Others, the changes of distance between the target and the optical center. Therefore, I use 8 parameters to describe δ, they are S_x, S_y, θ_x, θ_y, θ_z, x, y and Z_z. Accordingly, the projective transformation δ can be defined

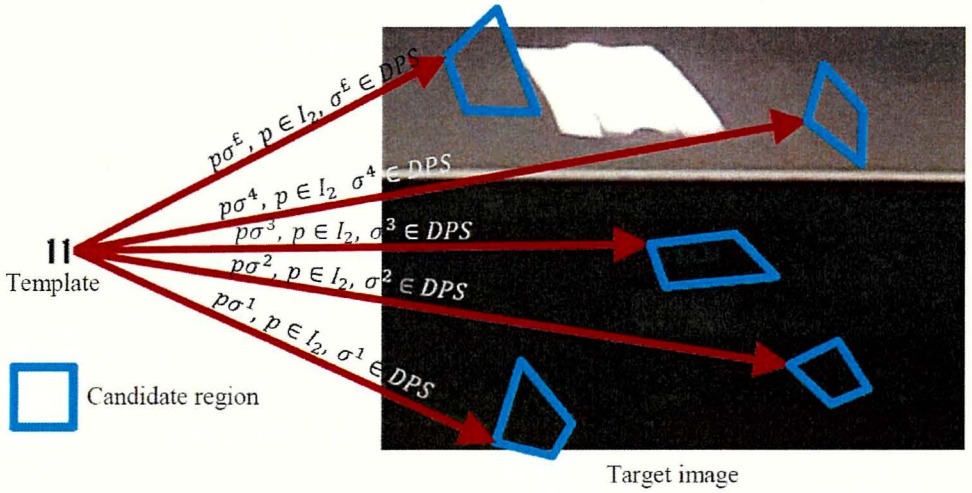


Figure 4.1: Candidate regions are shown by blue box on target image. Each candidate region is mapped to target image by homographic δ^{τ}.

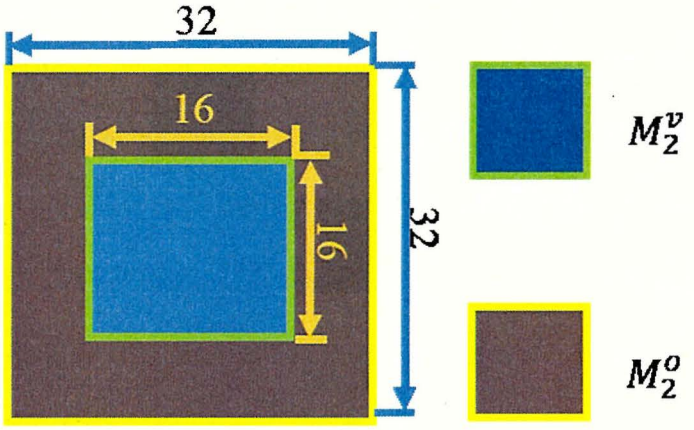


Figure 4.2: Without character information template M_2. M_2 also can be divided into two parts, VIS region M_2^v, and the front galss around the VIS region M_2^o. The position of M_2^v is in the center of M_2.

as formula 4.2:

$$\delta = \begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos\,\theta_x & sin\,\theta_x & 0 \\ 0 & -sin\,\theta_x & cos\,\theta_x & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} cos\,\theta_y & 0 & -sin\,\theta_y & 0 \\ 0 & 1 & 0 & 0 \\ sin\,\theta_y & 0 & cos\,\theta_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} cos\,\theta_z & sin\,\theta_z & 0 & 0 \\ -sin\,\theta_z & cos\,\theta_z & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ x & y & 0 & 1 \end{bmatrix} \times \begin{bmatrix} Z_z & 0 & 0 & 0 \\ 0 & Z_z & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & Z_z \end{bmatrix} \tag{4.2}$$

There parameter θ_x, θ_y and θ_z are rotation angles with corresponding to each axis. Parameter x, y are the translation size with respect to x, y axis on the target image plane. Parameter S_x and S_y are scale size with respect to each axis. Parameter Z_z is the distance between target and optical center, which the effect of Z_z is different in image size. However, I know that image size can be controlled by S_x, S_y. Therefore, Z_z can be fixed on a constant. With the δ defined, the p^δ can be calculated by multiply the matrices:

$$p^\delta = p\delta \quad p \in R^{(1\times4)}, \delta \in R^{(4\times4)}. \tag{4.3}$$

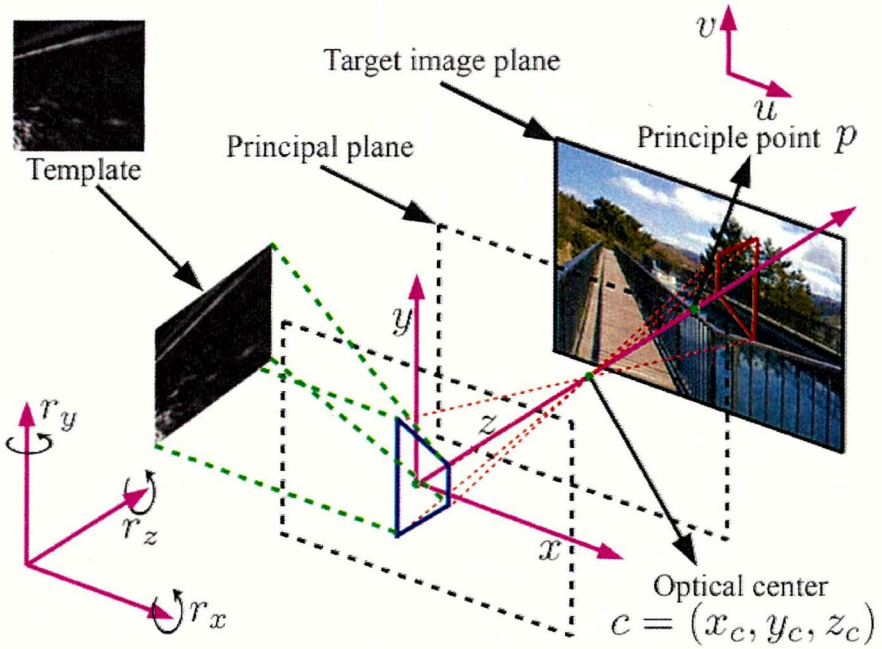


Figure 4.3: Pin-hole camera model, From the model we can know that the 2D projective transformation can be viewed as a transformation within the 3D and then projected onto a 2D plane. It is comprising by eight simple transformations on 3D.

Here the p is defined as $p = (p_x, p_y, 0, 1)$. In this method, when the coordinate of p^δ is calculated, the value of four dimensions should be normalized to one.

4.2.5 RBSM for VIS

In this method, three constraints are designed according to the inherent pixel distribution of the VIS area and the area around the VIS. Firstly, the pixel value distribution of the VIS region is not flat, because the VIS is the green paper with written back. In this method, I utilize the pixel values at the highest a percent to approximate the pixel values of VIS background in M_2^v for every candidate region. And utilize the pixel values at the lowest b percent to approximate the pixel values of the VIS text region. Where $a, b \in R^+, a + b \leq 1$. Secondly, the pixel value distribution between the sticker background region and glass region is not flat. Thirdly, the glass around VIS is an approximate flat.

I degine a function $T(.)$ to measure the flatness degree of VIS:

$$T(I_1, M_2, \delta) = \frac{\overline{l^b_{p \in M_2^v}(I_1(p^\delta))}}{\overline{l^b_{p \in M_2^v}(I_1(p^\delta))} + \overline{u^a_{p \in M_2^v}(I_1(p^\delta))}}. \tag{4.4}$$

$l^t_{p \in S}$ is sum of the pixel value at lowest t percent in the set S. $u^t_{p \in S}$ is sum of the pixel value at highest t percent in the set S. From the section 4.2.5 analysis, I can know that lower score of $T(.)$ means a more similar between the candidate region with the VIS.

Besides, I define a function $H(.)$ to measure the degree unflatness between VIS

background region and glass region.

$$B(I_1, M_2, \delta) = \frac{\overline{\sum_{p \in M_2^o}(I_1(p^\delta))}}{\overline{\sum_{p \in M_2^o}(I_1(p^\delta))} + \overline{u^a_{p \in M_2^o}(I_1(p^\delta))}}. \tag{4.5}$$

As I can observe from above function, a lower score means a more adaptive constraint.

Moreover, a function $G(.)$ is defined to measure the flat degree of glass area.

$$G(I_1, M_2, \delta) = \frac{\overline{u^c_{p \in M_2^o}(I_1(p^\delta))}}{\overline{u^c_{p \in M_2^o}(I_1(p^\delta))} + \overline{l^c_{p \in M_2^o}(I_1(p^\delta))}}, \ c \in R^+, c \leq 0.5. \tag{4.6}$$

In the result, the lower score of $G(.)$ means that the glass area is more flat. With function $T(.)$, $B(.)$ and $G(.)$ defined, I can define F_2 as

$$\begin{aligned} F_2(I_1, M_2, \delta) = w_4 \times T(I_1, M_2, \delta) + w_5 \times H(I_1, M_2, \delta) \\ + w_6 \times G(I_1, M_2, \delta), \ \ w_4, w_5, w_6 \in [0, 1]. \end{aligned} \tag{4.7}$$

Where w_1, w_2 and w_3 are the weights within $[0, 1]$. The experiment will be introduce in next.

4.2.6 Optimization for VIS

Coding of projective transformation parameter

For detecting the VIS, there still exists a problem that is continuous parameter space of projective transformation PS corresponding to infinite candidate regions. It is impractical from the infinite candidate regions to researching the optimum region. To solve this problem, a finite discrete set is extracted from PS by uniform step, in

which the number of each parameter is 2^n. The discrete set corresponding to a finite candidate region space is noted as DPS. When the finite candidate space is larger enough, then the optimum solution δ^{ψ} in DPS is very close to the optimum solution δ^*. Through experiments, when n is 8 the accuracy of the result is satisfactory. In order to search for the optimum solution in the DPS, all possibilities of DPS is needed to code into the chromosome. Accordingly, each parameter of δ is coded in an 8-bit binary.

Searching optimum region

According to section 4.2.6, the problem converted to that from the finite discrete set DPS to find an optimum solution. There still exists a problem that the DPS is massive. It has 2^{56} possibilities. This causes testing the complete discrete candidate space difficultly. In order to overcome this problem, I use a level-wise adaptive sampling (LAS) algorithm [23] to evaluate the approximate optimum solution. The flow chart of this algorithm is shown in Fig.4.5. Next, I will introduce this algorithm and analysis the advantage of this algorithm for this method.

Initialization

In this algorithm, the initial generation is noted as P^0, it includes n individuals. Each individual chromosome refer to a candidate region, Each individual chromosome generates by random 56 binary genes. Each individual chromosome corresponds with randomly parameters of projective transformation DPS.

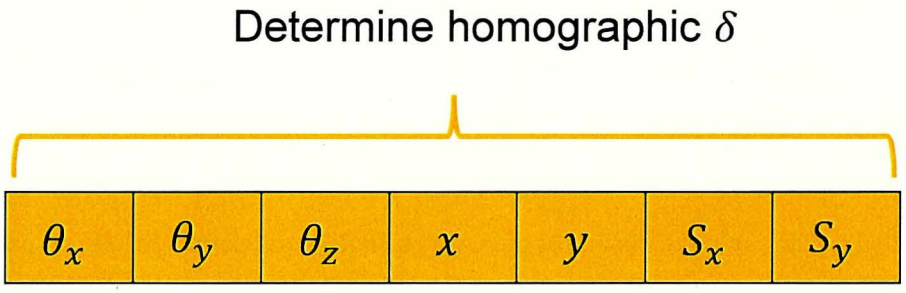


Figure 4.4: Chromosome. Yellow part are genes of candidate region, green part are genes of template background pixel value, each paramter coded by 8 bit gene.

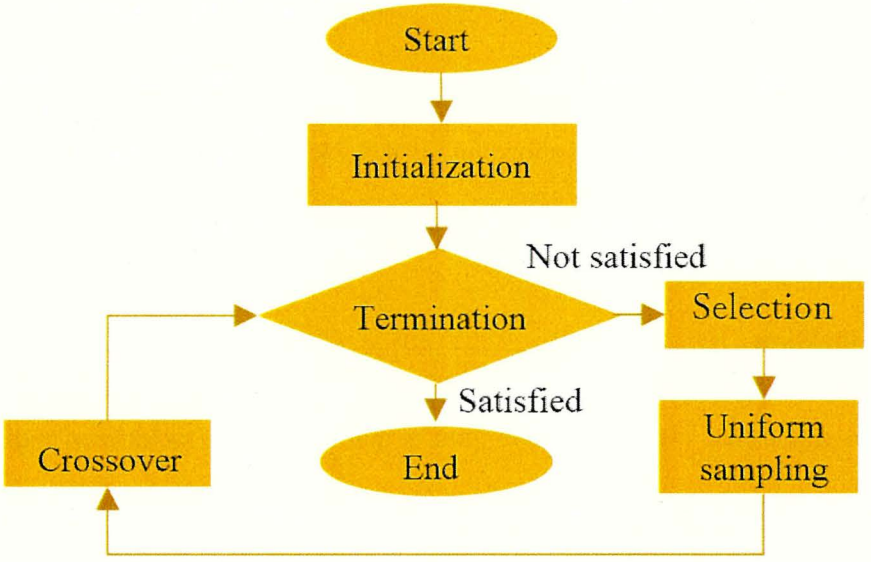


Figure 4.5: level-wise adaptive sampling algorithm flow chart.

Selection individuals with level-wise adaptive

In this section, According to the fitness value, some individuals are deleted. The specific selection method is as follows. Firstly, based on the chromosome, calculate each individual of generation P^{3m} correspond parameters δ and σ, where m is the generation number. Then according to the δ and σ calculation the fitness value to each individual. Finally, delete some individuals of which fitness value are smaller than the threshold TH_m in generation P^{3m}. Especially, the TH_m is level-wise adaptive, every time deleted individuals is close to a fixed proportion of the current generation, where the proportion in the range $[d_l, d_h]$. The remained individuals form a temporary generation of P^{3m+1}.

The TH_m is level-wise adaptive and the massive number of individuals results in a new problem, deciding the TH_m value is difficult. To solve this problem, in this algorithm, a stepwise approximation method is used, this method based on probability and statistics theory utilizes some random sampling to estimate the fitness value distribution of the whole generation. The method is as follows:

Step1: Initial the TH_t as the fitness value of the optimum individual in the current generation.

Step2: Randomly sampling an individual set that has T individual from the current generation, the random set is noted RS.

Step3: calculate the proportion PT that is individual' fitness value more than the TH_t in the RS. Then put PT into the equation 4.8:

$$TH_t = \begin{cases} TH_t \times 1.1 & , \quad PT < d_t \\ TH_t \times 0.9 & , \quad PT > d_t \\ TH_t & \quad others \end{cases} . \tag{4.8}$$

Step4: Repeating the step2 to setp3 until the PT belong the range $[d_l, d_h]$. The threshold TH_t is outputted threshold TH_m. The method of level-wise adaptive choosing individuals suits our method very well. The distribution of fitness value is different in the different environments, the fitness values may be concentrated in a small range in a generation. If an fixed threshold is utilized to select the individuals, the run time and the result accuracy is out of control. Thus, the level-wise adaptive selection method is selected to delete individuals.

After deleted some individuals, a uniform sampling method is utilized to select the next generation of individuals. Firstly, the range of fitness value in P^{3m+1} is evenly divided into ϕ range. Then, the same number of individuals is randomly extracted in each range. The extracted individuals form a temporary generation of P^{3m+2}. And the number of P^{3m+2} individual is the same with the P^{3m+2}. It is noteworthy that an individual can be extracted many times.

The similarity evaluation function 4.7 include three-part rules. It is possible that there is a candidate region where a part rule or two-part rules has a high similarity with VIS, but another part is not, which will cause a problem that the algorithm may fall into a local optimum solution. The operation can improve gene diversity, which is conducive to escape from local optimums. It has a great significance to this method.

Crossover

The uniform crossover operation is used to increase the diversity of chromosomes. The operator is shown in Fig.5. Parents are randomly generated from generation P^{3m+2}. And the children chromosome is obtained by mixing parents gene, the mixing ratio is mr. As a result, the children's chromosomes inherit the parent gene. And a child chromosome inherits approximate mr genes of the second parent, another inherits the genes of the first parent. Especially, the operator is not for all the individuals in P^{3m+2}, it only operates randomly rc percentage individuals. The operation results form a new generation P^{3m+3}, and I note it is generation m+1.

Termination and output

The selection of individuals process to the crossover process is repeated until the termination condition has been reached. In the algorithm, the terminating condition is that the number of individuals of m generation is smaller than g. The best homography $\delta^{\sim}$ with minimum fitness value is outputted.

4.2.7 Experiment

Datasets

In order to evaluate the RMSM for VIS, three datasets are collected in a different environment. These datasets are utilized to evaluate the performance in different aspects.

Dataset D1 is taken in the garage with 10 images, which the environment changes include illumination and position relationship between target and camera. And in

dataset D1, all of the vehicle inspection sticker the month is November. In addition, in order to more comprehensive check, the robustness of the proposed method Gaussian noise is added. The dataset D1 utilize to evaluate proposed methods can location the VIS in projective space.

Dataset D2 includes 527 images, which took in the following environment. The camera is fixed on a moving car, and the distance between the camera and VIS is 20 centimeters. And the car is driven through different places. And the month on the vehicle inspection sticker includes January and November, and the D2 includes 327 images of November and 200 images of January. This method utilizes The dataset D2 utilized to evaluate the performance for reflection.

Moreover, 50 images are taken in the gas station to construct a dataset D3. These images are taken at night, the camera position is fixed and the height of the camera position is 2 meters. Then, the car is moving in the direction of the camera. This dataset is utilized to evaluate the practical performance of the proposed methods. And for every image, the ground truth is demarcated by the manual.

Result evaluation

For evaluate the performance of RMSM for VIS, the overlap rate between ground truth and our result is utilized. When the overlap rate is more than a threshold, that is judged the localization success, The threshold as th. The determination method is as following,

$$result = \begin{cases} true & , \text{ if } overlaprate > th \\ false & , \quad others \end{cases} . \tag{4.9}$$

Table 4.1: List of each projective transformation parameter's range, the step size and amount in sampling set.

Parameter	Range	Step amount	Step size
θ_x	$[0, 0.3\pi]$	2^n	$0.3\pi/2^n$
θ_y	$[-0.05\pi, 0.05\pi]$	2^n	$0.1\pi/2^n$
θ_z	$[-0.05\pi, 0.05\pi]$	2^n	$0.1\pi/2^n$
x	$[0, n_1]$	2^n	$n_1/2^n$
y	$[0, m_1]$	2^n	$m_1/2^n$
S_x	[1.0, 3.0]	2^n	$2.0/2^n$
S_y	[1.0, 3.0]	2^n	$2.0/2^n$
Z_z	-	-	-

Parameter setting

In the experiment, the projective transformation parameters area set as Tab 4.1. Others, the initial generation number n is set as 150,000, the last generation number g is 1,000. The threshold TH_m is that every time eliminate 5% to 10% of last generation.

Moreover, according to the real VIS the shape of template is set as follows. The background of VIS a is 70% and the text area of VIS b is 15%, which the rest 15% is an uncertain area. The glass of around VIS top and low part g set as 20%. It is noteworthy that, in different experiments the fitness functions parameters are different.

Results

In the following experiment, the overlap rate threshold is set as 0.5, when the overlap rate is more than 0.5, the location result is judged as a success. When the weight of fitness function are that $w_4 = 1$, $w_5 = 1$ and $w_6 = 1$, the IRBM for the dataset D1 the success rate is 90%, the example of results is shown in Fig. 4.7.

When the weight of fitness function are that $w_4 = 0.5$, $w_5 = 0.5$ and $w_6 = 1$, the IRBM for the dataset D2 the success rate is 94.1%, the example of results is shown in Fig.4.8. The Fig.4.8 and the numerical results indicate that the IRBM has robust performance against reflection.

Moreover, the dataset D3 is used to evaluate the practical ability of IBRM. The experimental position is the gas station, when the vehicle is refueling, the vehicle position is approximately fixed. Thus, in this experiment, the region of interest (ROI) is utilized to instead target the image. Where the ROI is fixed to the middle one-third of the target image, shown in 4.9 the blue rectangle. When the weight of the fitness function is that $w_4 = 0.5$, $w_5 = 1$ and $w_6 = 1$, the success rate is 92%, which the IRBM for the dataset D3. the example of results is shown in fig. 4.8. And the last three images of D3 are blurred images, shown in location false image in Fig.4.9. When the dataset D3 exclude the last three image, the success rate is 98%. The high success rate indicates that the proposed method IBRM is can be used at the gas station for locating the VIS.

4.3 Rule-based matching for RFP detection

4.3.1 Background

To save on the human labor costs, I herein plan to implement an automatic canning robot for packaging roast fish from the wire mesh belt conveyor (WMBC) line. One of the principal challenge that must be addressed is the development of a machine vision system. The roast fish parts have the following features. All parts have various patterns, size, shape, color, and color combination. Also, the size of RFP is

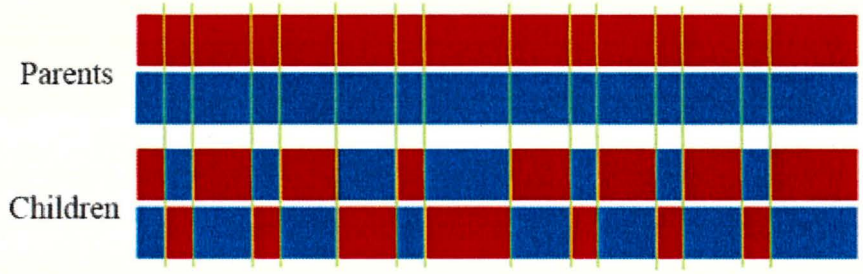


Figure 4.6: Uniform crossover operator.

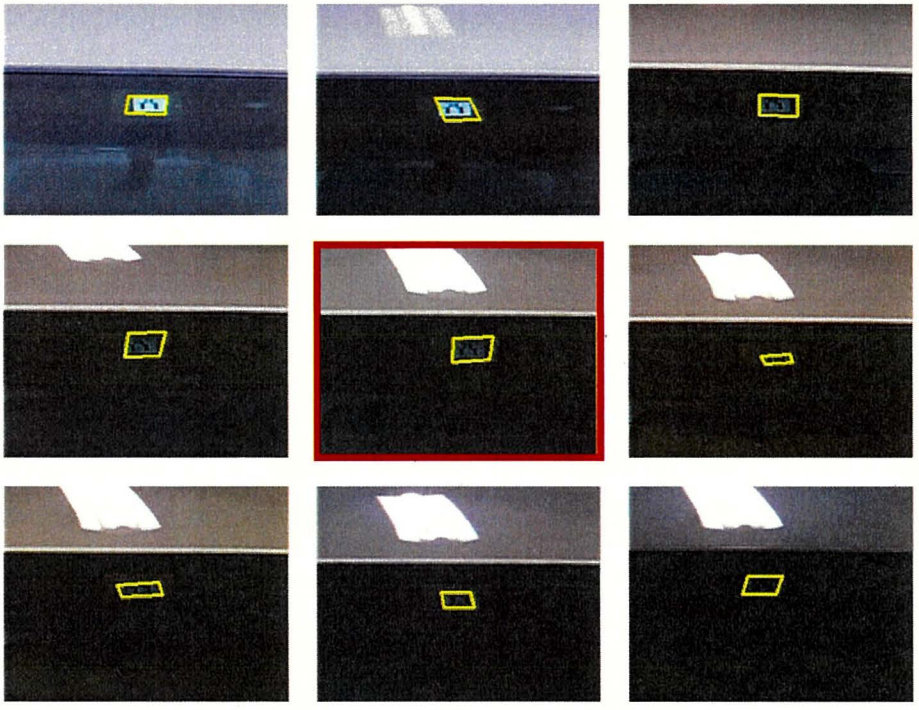


Figure 4.7: Visual results of RBSM for dataset D1. Localization results is represented with yellow bounding box. The red rectangle marks the error results.

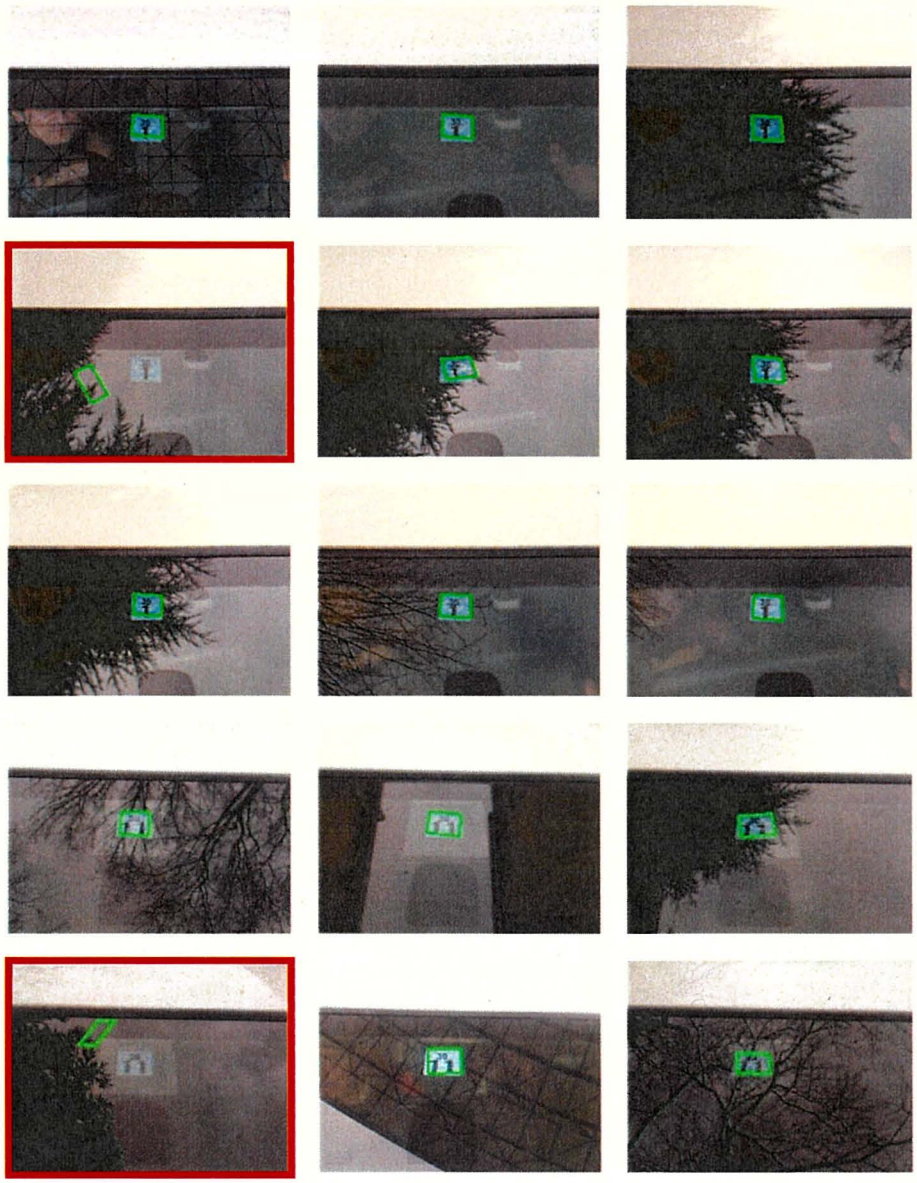


Figure 4.8: Visual results of RBSM for dateset D2. Localization results is represented with green bounding box. The red rectangle matks the error results. These images indicate that the IRBM has robust performance for different VIS.

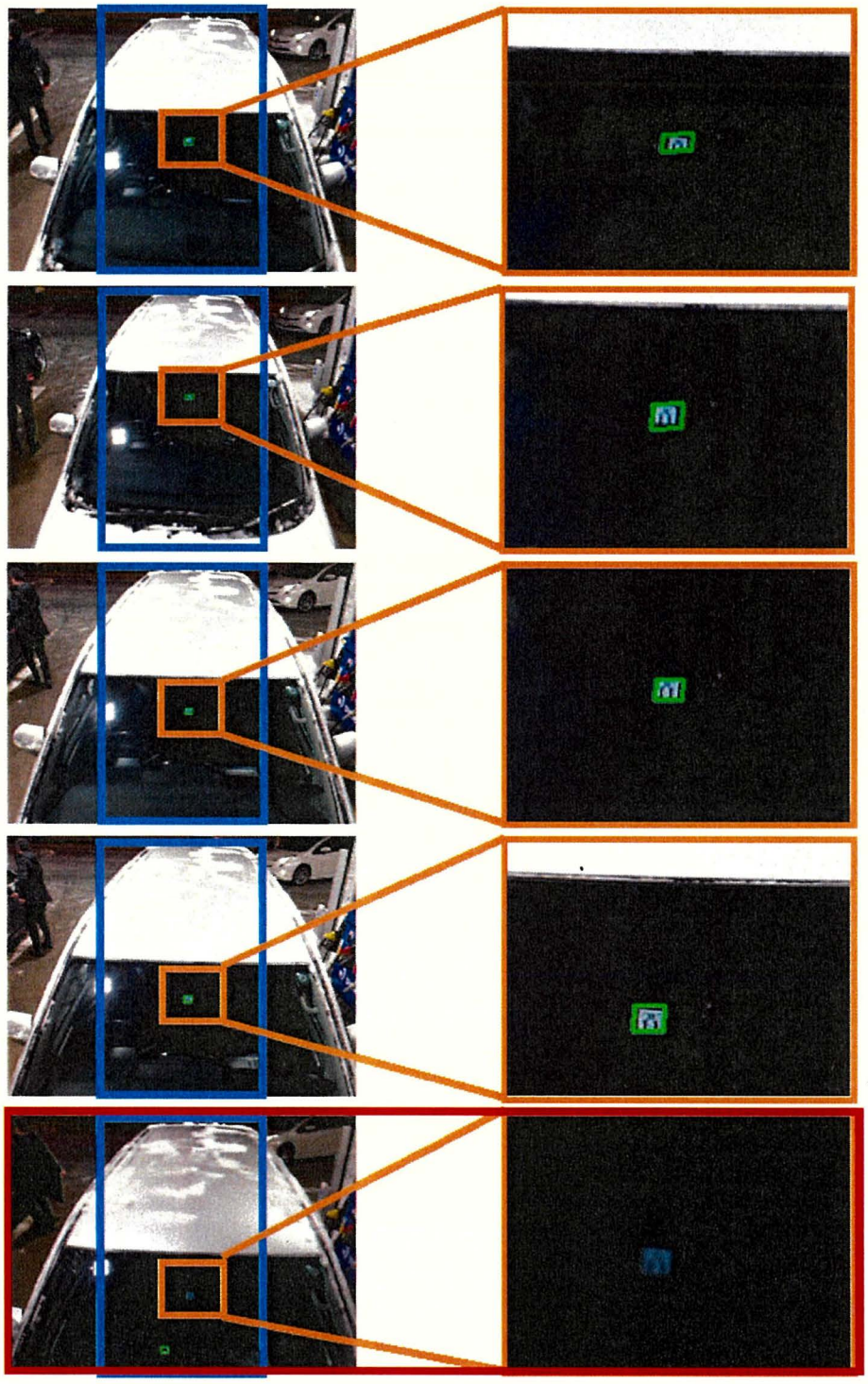


Figure 4.9: Visual results of RBSM for dataset D3. Localization results is represented with green bounding box. The red rectangle marks the error results. These images indicate that the RBSM has robust performance for different VIS.

bigger and the shape is more unified than the tail part. To guide the robot arm while canning roast fish, the machine vision system must meet the following requirements. (1) All parts are the canning material. This system must sense the angle and position of all parts to guide the robot arm. (2) To make the product more attractive, the bigger and more unified fish part should be put on the top. For this canning rule, the system needs to be distinguished two kinds of fillets, because of the feature of fish parts. (3) Because there are various production environments, the illuminations are different. Furthermore, there could be some shadow caused by the staff or machine working site lead to that the system should be robust to the level of illumination. And in this thesis, only the RFP detection is introduced, owing to the tail part is deal with by another method. And the RFP is noted as the roast fish part (RFP).

There are some difficulties to develop a vision system, which satisfies the above requirements. First, the size, shape, and color are not exactly the same within the same type. Second, the connected objects make it extremely difficult to detect, because some objects are put very close (Fig. 3.10(a)). The uncertain number of objects is also challenging, making it more difficult.

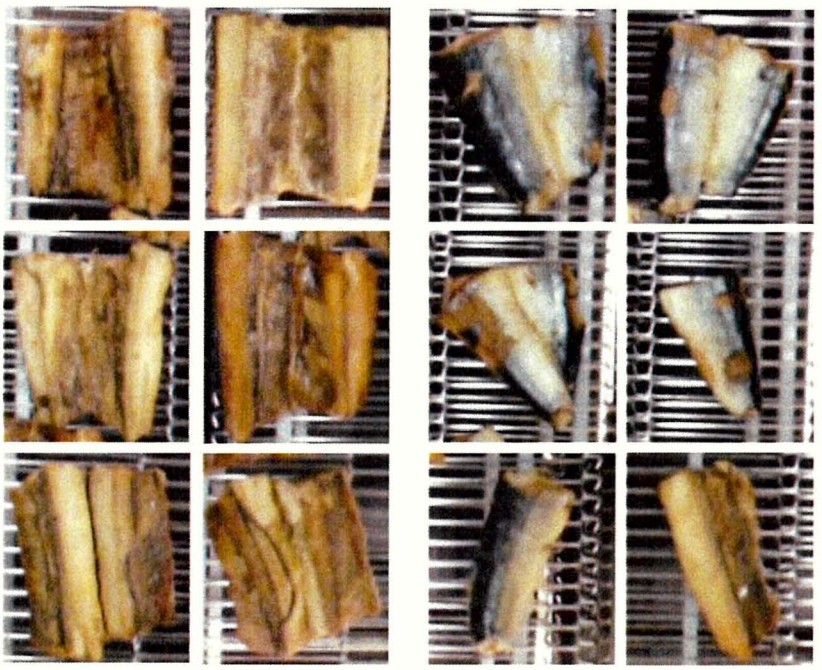
4.3.2 Problem description

To develop a robust-assisted packaging system, which can guide the robot arms to pack the roast sauries into cans, it needed to detect the roast sauries part. For gripping strategy generation, the system is required not only to be able to detect the roast saury area but also to estimate the geometric parameters. Besides, according to different canning requirements, it is also necessary to distinguish the type of fish parts. In this thesis, a rule-based matching method is utilized to detect a kind of fish

part. And the left part can be detected by other methods, such as some segmentation method [74]. And in this thesis, only the rule-based matching method is introduced. The example is shown in Fig. 1.3.

A target image is given, noted as T. And T can be divided into two parts. one is the background. The other part is the roast fishes. And as illustrated in 4.10, the roast fishes include two kinds of fish part. body and tail parts, that have the following features. All parts have various patterns, sizes, shapes, colors, and color combinations. Also, the size of the RFP is bigger and the shape is more unified than the tail part. The RFP is noted as $b \in B$. The number of elements of the set B can be zero or more. Our purpose is to detect all elements in B.

In the RFP detection, the input is the gray-scale ROI image I_g and the output of the object region O. For the better quality of products, the RFP will be put facing the skin surface to the camera. As shown in Fig. 4.10(a), RFPs have two kinds of patterns: white-black-white (WBW) and black-white-black (BWB). The black region is due to the region of the body with some internal organs and blood, as this part will become black after roasting. Differences during cutting then lead to these two different patterns. The various shape, sizes, and colors of RFPs lead us to develop an algorithm to deal with the challenge of detecting multiple RFPs with two patterns. To solve this problem, a rule-based multi-object matching method is introduced. Firstly, the common features of RFPs are utilized to design some rules and a supporting template. These rules are utilized to measure the probability that the candidate is the RFP. Then, a mathematical model is used to obtain the candidate regions via template mapping. Finally, a GA is used to search for the local optimal solution by introducing DCAPD for multi-object detection.



(a) (b)

Figure 4.10: Sample fish parts. (a) Some samples of fish pieces are considered to be the RFP. These samples can be divided into two-part white-black-white (WBW, first and two rows) and black-white-black (BWB, the last row) patterns. (b) Some samples of fish pieces are considered to be tail parts. These tail samples have large differences in size, shape, and color. The third row of the second column sample is a broken RFP and is viewed as a tail part.

4.3.3 Flexible Template

Although the RFPs are different in shape, size, and color, they still have some commonalities. The shape of the RFPs is an approximate rectangle. Furthermore, according to the color, the two patterns of fish RFPs can be divided into three parts. However, as shown in Fig. 4.10(a), the shapes, and size of each part is not fixed. To solve this problem, the fuzzy field is introduced to the template. Therefore, the template is shown in Fig. 4.11(a), denoted as T, with size 50×50 pixels. This template includes following four regions, they are the left end T_1, right end T_2, middle T_3, and T_4 which is the boundary of T_3 with T_1 and T_2, the sizes of T_1 and T_2 are 8×50 pixels. The size of T_3 is 16×50 pixels, and each part of T_4 is 9×50 pixels. The T_1 and T_2 refer to the black region in the BWB pattern or the white region in the WBW pattern, T_3 refer to the white region in the BWB pattern or the black region in the WBW pattern, and T_4 can be a mixture of black and white. Results in the T_4 part of this template can be used to account for variations in the RFPs.

4.3.4 Candidates

With the template devised, the candidate regions that may exist RFP can be evaluated using the template with some geometric transformations to map to the target image. As shown in Fig. 3.10(a), these geometric transformations include scaling, rotation and translation. The following five parameters are used to describe these transformations. (1) Scaling parameters for the x and y axes S_x and S_y. (2) The angle of the center rotation θ. (3) Parallel translation for two axes x and y. A point $p(i, j)$ in the template can be mapped to a point p^{τ} in the target image via the above

transformation, such that p^τ can calculated by the following formula:

$$p^\tau = \begin{pmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} cos\theta & -sin\theta & 0 \\ sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 0 & x \\ 0 & 1 & y \\ 0 & 0 & 1 \end{pmatrix} \tag{4.10}$$

And every point in the template T via a τ mapping to the target will result in a candidate c^τ. An example of c^{τ_i} is shown in Fig. 4.13.

4.3.5 Pretreatment

The region of interest (ROI) is fixed in the target image, with a size of $n_1 \times m_1$ pixels. This ROI is denoted as I, as shown in Fig. 3.10(a). The ROI can be divided into three kinds of regions according to the WMBC position. They are object region O, WMBC region W, and the background region B. The segmentation of W is due to some significant features that can be used, such as the edge and shape. Subsequently, if B can be segmented, then O is obtained by I rid of W and B.

To segment W, the raw image I is converted to a grayscale image, denoted as I_g. The gradient feature is utilized to extract the edge of the WMBC as follows. First, for every point, the gradient is calculated using the Sobel operator in the horizontal and vertical directions denoted as $\vec{r}$ and $\vec{c}$, respectively. Then, add up the two-directional gradient vectors to form the gradient vector $\vec{s}$, as shown in Fig. 4.12. Next, the magnitude $|\vec{s}|$ and angle $\langle\vec{s}\rangle$ of $\vec{s}$ is calculated for every pixel. Finally, based on the following two conditions judge whether the pixel is in the W. The value of $|\vec{s}(i,j)|$ must be sufficiently large, on the other hand, the direction of $\vec{s}$

must be closed to the y axis. The analysis results in pixel values exhibit only a small change inside W, but a dramatic change at the boundary between W and B or O. Formally, Eq. 4.11 is defended to obtained W.

$$W(i,j) = \begin{cases} 1 & \text{if} \begin{cases} \langle \vec{s}(i,j) \rangle > (90° \times t - \theta_{th}), \\ \langle \vec{s}(i,j) \rangle < (90° \times t + \theta_{th}), \\ t \in \{1,3\}, |\vec{s}(i,j)| > \overline{|\vec{s}|} \end{cases} \\ 0, & \text{otherwise.} \end{cases} \tag{4.11}$$

In the Eq. 4.11, θ_{th} is the threshold, which can control the segmentation results. When the θ_{th} is larger, the region is easier to be segmented as O, and vice versa. The $\overline{|\vec{s}|}$ is average value of gradient magnitude in I. The approximate W can be obtained by the above method, and an example result is shown in Fig. 3.10(b).

The B is divided into some narrow parts by the horizontal WBMC. Therefore, a vertical filter is used to check W and determine B, if a region of length without the wire mesh is shorter than the threshold TH_v, this region is determined as B. The TH_v is defined by the gap of WMBC in the target image. The remaining region is determined as the object region O. In our case the over-segmentation (W or B is segmented as O) is acceptable due to the following processing can deal with this problem. However, the under-segmentation (O is segmented as W or B) will give a negative effect for the following processing. The above parameters can be utilized to avoid under-segmentation.

4.3.6 RBSM for RFP

With the candidates defined, our problem becomes searching for the candidates of the RFP. However, it is a challenge that the traditional pixels-based measurement methods, such as SSD, SAD, and BBS, are ill-suited to measuring the fitness of a candidate for the RFPs because these methods need varied templates to identify varied RFPs. In this work, the regional differences in gray scale value are utilized to devise two rules for evaluating the candidate. These rules rely only on the information in the candidate image. The first rule is that the gray scale value must be different between T_1, T_2 and T_3 in a RFPs, and the average pixel values of T_1 and T_2 must be larger compared with that of T_3.

$$R_1(I_g, T, \tau) = abs\left(\overline{\sum_{p \in T_1 \cup T_2} I_g(p^\tau)} - \overline{\sum_{p \in T_3} I_g(p^\tau)}\right) \tag{4.12}$$

The second rule is that the gray scale value difference must be small between T_1 and T_2. Therefore, the average pixel values must be similar between T_1 and T_2.

$$R_2(I_g, T, \tau) = 1 - abs\left(\overline{\sum_{p \in T_1} I_g(p^\tau)} - \overline{\sum_{p \in T_2} I_g(p^\tau)}\right) \tag{4.13}$$

Furthermore, except for the gray-scale value, there is another cue for measuring candidates' fitness: the candidate must lie in the object part O. Accordingly, the formula 4.14 is used to apply this rule.

$$R_3(I_g, T, O, \tau) = \frac{\sum_{p \in T} \lfloor p^\tau \in O \rfloor}{\sum_{p \in T} \lfloor p^\tau \in I_g \rfloor} \tag{4.14}$$

The $\lfloor \cdot \rfloor$ is an indicator function that turns true and false into 1 and 0. This rule includes an implicit condition that the candidate is an rectangle. With above three rules are defined, the measure function can be given as follows.

$$f(I_g, T, O, \tau) = (w_1 \times R_1 + w_2 \times R_2) \times R_3 \tag{4.15}$$

The terms w_1, w_2 are the weights of rules within the range of $[0, 1]$. And a larger f value means the candidate referred to τ has a higher probability to be a RFP.

4.3.7 Searching RFP for all local optimal solutions

In the RFP detection, with candidates and matching rules defined, the problem becomes searching for suitable solutions overall candidates. However, there are still some problems when searching for multiple RFPs. Firstly, there are five parameters and an enormous number of candidates. When the searching gap is small for the candidate, the search is time-consuming, but increasing the searching gap will lead to reduced accuracy. Furthermore, multiple objects and the uncertain object number means that typical optimization method, such as GA and particle swarm optimization [75] (PSO), are difficult to be applied to our case. A special GA, that introduces deterministic crowding of the population for the five parameters being optimized [28], is used. This method is a kind of evolutionary method. It can use a small sample candidate set to search for a high accuracy approximate solution.

The searching algorithm is shown in Algorithm 1. First, which N parents are generated and each has randomly assigned values of the five parameters. Second, fitness f is calculated for each parent. Then, the parents are checked to see if some

individuals meet the conditions for creating a new cluster. These conditions include the following three items. (1) There exists an individual τ^{p_i} such that f is above the threshold th. (2) This individual does not belong to any cluster. (3) The distance between τ^{p_i} and each cluster above the threshold d_t. A cluster is constructed by a region and some individuals, where the region is a circle, whose center is the center of τ^{p_i} and whose radius is threshold d_t. Third, the children S is generated by selecting parents P for crossover and mutation. Fourth, the fitness is calculated for each child. Fifth, the children are compared with their respective parents and if the f of the children is larger than that of the parents, then the children are used to replace the parents. We repeat the processing from the second to the fifth until the termination condition is met. Finally, the optimum solutions of each cluster in the final generation are our results.

4.3.8 Experiment

Dataset

The proposed system is tested with the experimental system which simulated the real factory environment. The target image is taken by the camera in the box, as shown in Fig. 3.10(a). And we manually fixed an ROI for analysis, the size of which is 320 $\times$ 480 pixels, while the real working region size is 300 $\times$ 450 millimeters. Furthermore, an efficient range is utilized to prevent part of the object outside the ROI region, which is the center of ROI with the size of 320 $\times$ 380 pixels. The performance of the proposed system is evaluated by two original sets of data. In dataset 1, the object distribution is scattered and regular in each image. Dataset 1

Figure 4.11: Template and two pattern samples, (a) is the template. It can be divided into four parts. (b) is the examples of BWB pattern part. (c) is the examples of WBW pattern part.

Algorithm 1 DCAPD for RFPs detection

Input: input parameters T, Q
Output: Optimum solution of each cluster

Generate N parents $P = \{\tau^{p_1}, \tau^{p_2}, ..., \tau^{p_N}\}$ randomly
Calculate fitness f for each parent
while (Not termination condition) **do**
    if Meet the condition of create new cluster **then**
        Create new clusters
    end if
    Generate children $S = \{\tau^{s_1}, \tau^{s_2}, ..., \tau^{s_N}\}$ by parents
    Calculate the fitness f for each child
    Compare and replace the parents with children
end while
return Clusters C

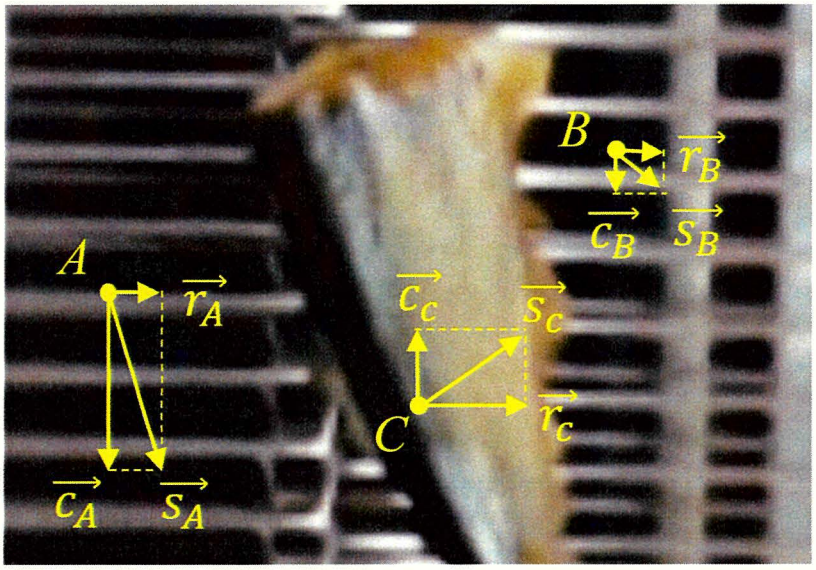


Figure 4.12: Example of horizontal wire mesh judgment. Point A indicates the wire mesh, and the magnitude of the gradient vector $\vec{c_A}$ that is calculated in the vertical direction Soble operator is large. Therefore, the magnitude of $\vec{s_A}$, which is the sum of $\vec{r_A}$ and $\vec{c_A}$, is also larger here than at other points, and the direction is close to the y axis. The points B and C are located in the background and object regions, respectively, and the magnitude of $\vec{s_B}$ and $\vec{s_C}$ are small, and its direction is irregular.

has 55 images that include 195 RFPs. Dataset 2 includes 45 images composed of 166 RFPs, and the distribution of the objects is close and irregular.

Results for object region estimation

To verify the effectiveness of the foreground segmentation method, we implemented two kinds of segmentation methods Otsu segmentation and background subtraction. And we employ the overlap rate (OR) between ground truth W_g and the segmented result W_r to measure the results, which is defined as: $|W_r \cap W_g| \,/\, |W_r \cup W_g|$. Here, the operator $|\cdot|$ is used to count the number of pixels within a set. The higher OR means that the result is closer to the ground truth. In addition, we use the under-segmented ratio of USR to evaluate the negative effect of under-segmentation. Which the OSR is defined as $|W_g - W_r \cap W_g| \,/\, |W_g|$. The USR is the error ra-

Table 4.2: Results for object region estimation

Method		Dataset 1	Dataset 2	All dataset
Our	OR	0.773	0.762	0.768
	USR	0.053	0.098	0.068
Background subtraction	OR	0.281	0.325	0.301
	USR	0.573	0.552	0.563
Otsu segmentation	OR	0.322	0.431	0.371
	USR	0.390	0.368	0.380

tio that foreground is wrongly segmented as background. The results are shown in Table4.2. The highest OR and lowest average MSR indicate that our method is the most effective with the least negative impact on the three methods.

Detection results

In RFP detection, the GA that introduces DCAPD is utilized for parameter optimization. The population size is set to 350, the crossover rate is 0.7, and the mutation rate is 0.05. In addition, to fix the running time, the termination condition is set as the generation number reaching 1200. The parameters of the transformation model are shown in Table 4.3, and each parameter is coded using 8-bit binary. To validate the superiority of rule-based matching compared with the commonly used pixel-based matching methods for a class of objects, we implemented two pixel-based measurement methods, the traditional SAD and the state-of-the-art BBS. These methods use two kinds of templates, that as shown in Fig. 4.11(b) and Fig. 4.11(c), running two times. Furthermore, for a fair comparison, we set the parameters of the optimization algorithm to be the same as in our methods.

Three important metrics is considered to evaluate detection results. One is the weight center deviation (WCD), that is, the weight center distance between ground

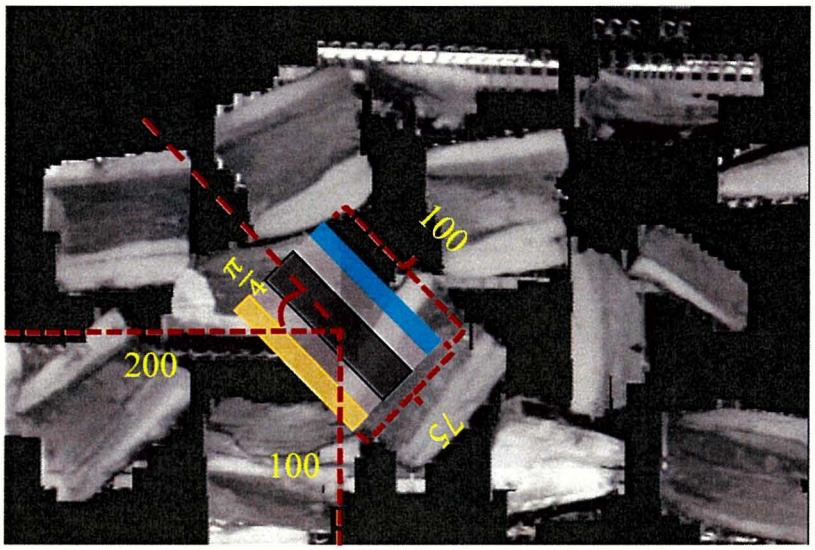


Figure 4.13: Example of a candidate. In this example, the translation length is $x = 100$ pixels, $y = 200$ pixels, the scaling size is $S_x = 1.5$, $S_y = 2$ and the rotation size is $\theta = \pi/4$. Notice that the figure is rotated $\pi/2$ counterclockwise to save space.

Table 4.3: Geometric parameters range

x	y	S_x	S_y	θ
[0,319]	[0,479]	[1.2,2.2]	[1.4,2.4]	[0,π]

Table 4.4: Results of RFP detection

Method	Dataset	Objects	TP	Error	Miss	AOR	AAD (degree)	AWCD (pixels)
Our	1	195	189	3	6	0.73	6.42	5.24
	2	166	158	2	8	0.73	6.82	5.00
	All	361	347	5	14	0.73	6.60	5.13
SAD	1	195	177	50	18	0.62	18.43	15.34
	2	166	136	39	30	0.60	19.79	16.50
	All	361	313	89	48	0.61	19.02	15.85
BBS	1	195	171	45	24	0.65	14.92	13.71
	2	166	139	36	27	0.62	16.49	15.67
	All	361	310	81	51	0.64	15.62	14.59
DS-SRI	1	195	165	27	33	0.65	12.78	12.51
	2	166	134	22	29	0.62	11.67	13.39
	All	361	299	59	62	0.64	12.29	12.90

truth and the results. After all the objects are detected, the robot arm sucks the center of gravity of the object with a suction cup to handing the object. Therefore, the WCD has a crucial effect on the grabbing success rate. The second metric is the angle deviation (AD), which is the absolute difference from the actual angle. In addition, the OR between results and ground truth is also utilized to auxiliary evaluate the results. The results are shown in Table 4.4, where the TP means the number of true positive (TP). Table 4.4 shows that our method performs our method is better than the others in TP, error rate, average WCD (AWCD), average AD (AAD), and average OR (AOR).

Furthermore, the success rate curve for the threshold of WCD is given in Fig. 4.14, Fig. 4.15, Fig. 4.16. The effective working range of the suction cup is a circle with a radius of 15mm. And in our implemented environment, one pixel is approximately one millimeter. Thus, the threshold of WCD threshold is set as 15 pixels, the success rate achieves 0.936. This high accuracy shows that our method is effective for detecting the RFP. Figure 4.14 demonstrates that the performance of BBS is better than that of the traditional methods SAD, but BBS is still significantly behind our method. And some results of examples are shown in Fig. 4.18, where the results also illustrate that the two patterns of RFP are detected significantly well. Moreover, DS-SRI also is employed to show the superiority of RBSM, the results are shown in 4.17. We can observe that the results of DS-SRI are the same as BBS. And the RBSM can outperformance these methods in RFP detection.

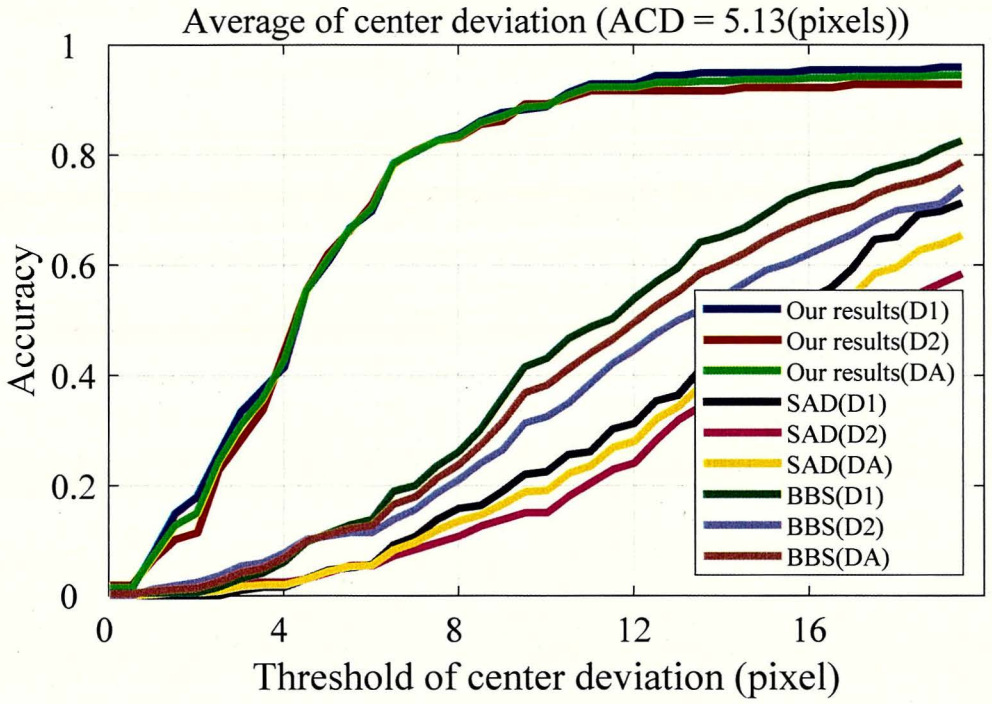


Figure 4.14: Success curve of RFP detection. This figure demonstrates three methods for the RFP detection: our method and two pixel-based methods, sum of absolute difference (SAD) and best-biddies similarity (BBS).

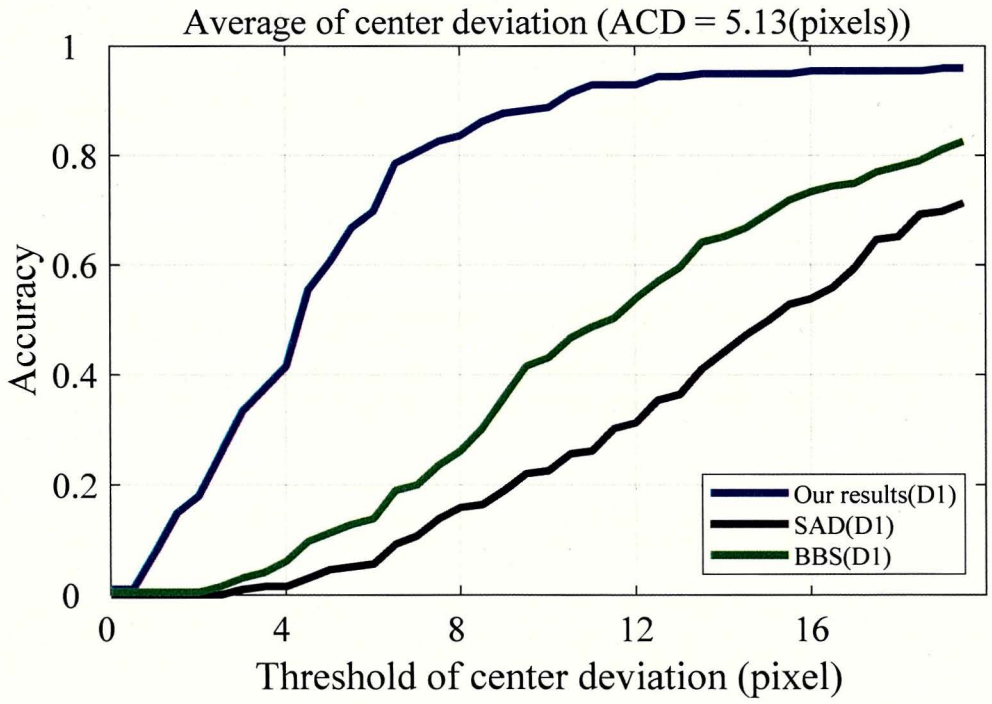


Figure 4.15: Success curve of RFP detection for dataset 1

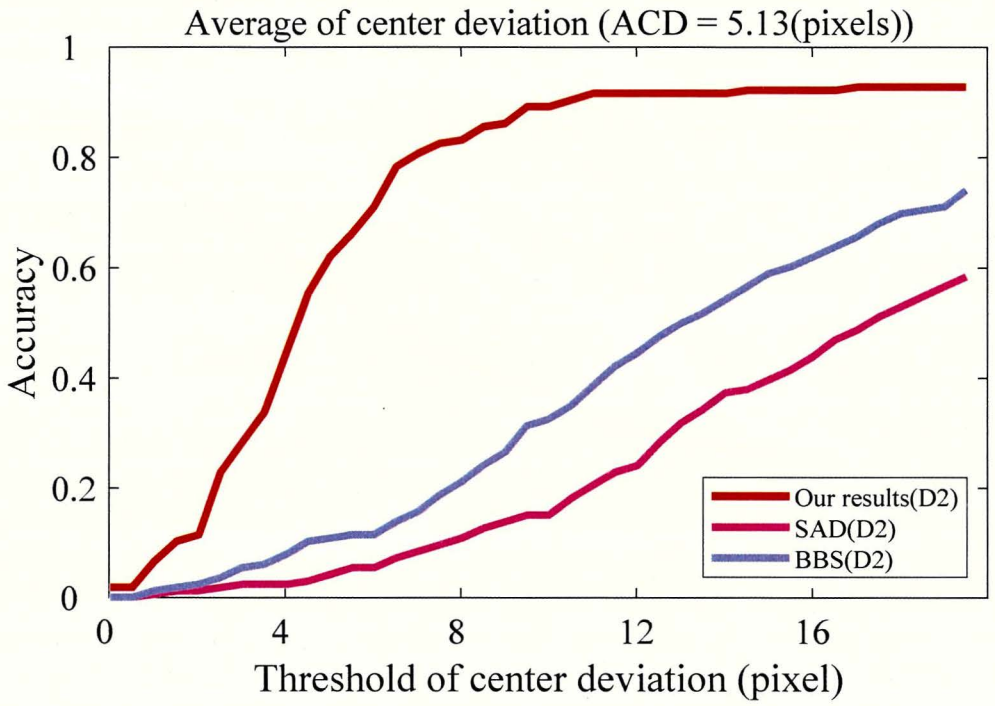


Figure 4.16: Success curve of RFP detection for dataset 2.

Table 4.5: Running time

	Pre	body			Proposed system
		Our	SAD	BBS	
Times(ms)	251	2237	2694	32638	2488

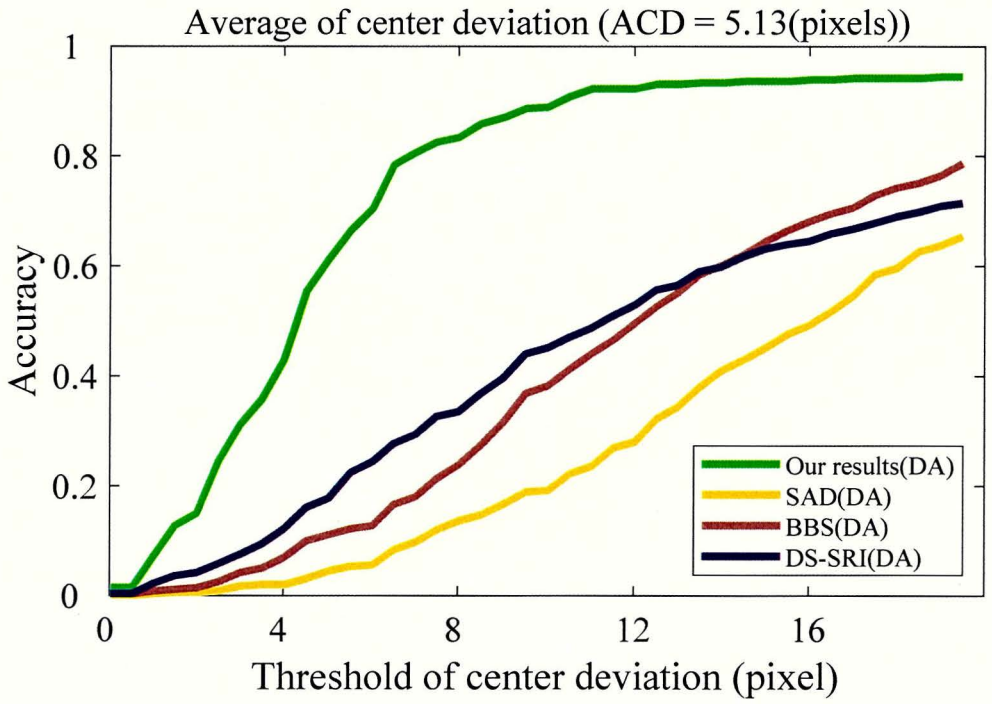


Figure 4.17: Success curve of RFP detection for all dataset. This figure demonstrates four methods for the RFP detection beside our method and two pixel-based methods, sum of absolute difference (SAD) and best-biddies similarity (BBS), DS-SRI all is employed.

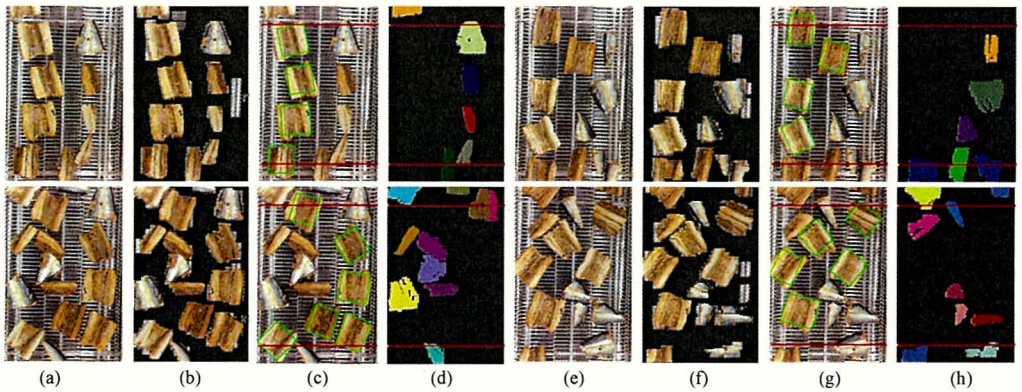


Figure 4.18: Visualization results. (a), (e) ROI of target images. (b), (f) Results of object region estimation, where the results demonstrate that almost all the background region is excluded, and the target region is well preserved. (c), (g) Results of RFP detection. (d), (h) Visual results of tail segmentation by efficient graph-based segmentation. In (c), (d), (e), and (f) the red line marked the efficient range.

Implementation environment and running time

This fish canning vision system is implemented using Visual Studio 2015 in Windows 10, and the running speed is about one second on a 2.7 GHz Pentium i7. The running time is given in Table 4.5. From this table, we can know that our rule-based similarity measurement method is close to the simplest similar measurement, SAD, and significantly faster than BBS. This method still is acceptable for our system and conspicuously faster than the mean-shift method. This system processes one image the average time is about 2.5s. This speed is fast enough for the fish canning robot because with an efficient working length of 380 pixels, about 356 mm in real space. In the real factory, the flow speed of WMBC is 2000mm per minute, and the efficient working range needs about 10.6 s. Accordingly, our processing speed is fast enough for the roast fish canning.

4.4 Conclusion

The rule-based matching method is proposed for a class of objects. In this method, the pixel distribution of the objects is utilized to design a template. Then, some rules are designed by the common feature of the objects. Finally, Some optimization algorithms are used to find targets from candidates.

To make our approach more intuitive, the rule-based method is introduced through two examples, which are rule-based matching for vehicle inspection sticker (VIS) detection and rule-based matching for roast fish part (RFP) detection. Firstly, a rule-based similarity method (RBSM) under the GA framework to solve the problem of locating the VIS region over the projective space. First of all, a template without

character information is made manually according to the real VIS and discusses its projective transformation. Then, according to the feature distribution of VIS, 3 rules are untitled to evaluate the similarity between candidate and VIS. Finally, level-wise adaptive sampling is applied. The results show that this method has a satisfactory performance under different environmental conditions.

The second piratical problem is RFP detection. The roast fish parts have a large difference. The color distribution can be divided into two patterns black-white-black and white-black-white. Moreover, the same pattern object also has some differences. According to the color distribution of objects, a flexible template is designed for two patterns. Then, the common features of objects are utilized to designed 3 rules. These rules are utilized to evaluate the candidate is the target or not. Finally, under the genetic algorithm framework combine with the deterministic crowding technique, deterministic crowding with adaptive population distribution (DCAPD) is utilized to search multi-object at the same time. The result showed that RBSM outperforms the other methods with adequate speed for the detection of objects.

The rule-based matching method can deal with multi-object at the same time, it can ignore some unneeded differences. With some specific template and rule, it can deal with the illumination change, rotation, and violently deformed. Moreover, this method can deal with various problems with various templates and rules. The most disadvantage is the rules are designed manually. Some objects may not have enough rules for objective evaluation.

Chapter 5

Conclusion and future work

In this thesis, a novel similarity metric based template matching method is proposed for object direction. The design of a good similarity metric is still difficult, because of the following problems, scaling, background occlusion, deformation, illumination change, multiple types of objects. Firstly, the diversity similarity measure against scaling, rotation, and illumination (DS-SRI) is proposed for single object detection. it takes advantage of the global statistic to deal with complex deformations, occlusions, etc. Extended bidirectional diversity combined with rank-based nearest neighbor search forms a scale-robust similarity measure, and the exploit of polar coordinate further improves the robustness against rotation. Moreover, in order to deal with the illumination change and further deformation, illumination-corrected local appearance and rank information are jointly exploited during the NN search. The experimental results have shown that DS-SRI can remarkably outperform other competitive methods.

Despite the robustness of DS-SRI, it still has a few limitations. It is likely to mislocate the object when the color distribution of the template is flat. It is also the

case when the patches in the template are similar to each other. Another it can not deal with the multiple object case.

Moreover, the rule-based similarity measure (RBSM) method is proposed to handle all problems, RBSM is proposed for a class of objects. In this method, the pixel distribution of the objects is utilized to design a template. Then, some rules are designed by the common feature of the objects. Finally, Some optimization algorithms are used to find targets from candidates.

To make our approach more intuitive, the rule-based method is introduced through two examples, which are rule-based matching for vehicle inspection sticker (VIS) detection and rule-based matching for roast fish part (RFP) detection. Firstly, a rule-based similarity method (RBSM) under the GA framework to solve the problem of locating the VIS region over the projective space. First of all, a template without character information is made manually according to the real VIS and discusses its projective transformation. Then, according to the feature distribution of VIS, 3 rules are untitled to evaluate the similarity between candidate and VIS. Finally, level-wise adaptive sampling is applied. The results show that this method has a satisfactory performance under different environmental conditions.

The second piratical problem is RFP detection. The roast fish parts have a large difference. The color distribution can be divided into two patterns black-white-black and white-black-white. Moreover, the same pattern object also has some differences. According to the color distribution of objects, a flexible template is designed for two patterns. Then, the common features of objects are utilized to designed 3 rules. These rules are utilized to evaluate the candidate is the target or not. Finally, under the genetic algorithm framework combine with the deterministic

crowding technique, deterministic crowding with adaptive population distribution (DCAPD) is utilized to search multi-object at the same time. The result showed that RBSM outperforms the other methods, that include the DS-SRI, with adequate speed for the roast fish part detection.

The results of these practical problems illustrate that RMBS can cover all the above difficulties with suitable templates and rules. The rule-based matching method can deal with multi-object at the same time, it can ignore some unneeded differences. With some specific template and rule, it can deal with the illumination change, rotation, and violently deformed. Moreover, this method can deal with various problems with various templates and rules. The most disadvantage is the rules are designed manually. Some objects may not have enough rules for objective evaluation.

In the future, I will focus on developing an automatic method for multiple types of the object detection method. Specifically, templates and rules are automatically designed for the RBSM method. More details. the objects are divided into some super pixels automatically, and the rules are also automatically designed according to some training data.

Acknowledgement

I would like to express my deepest gratitude to associate Prof. Takuya Akashi for his excellent guidance, patience, endless support, and for providing me a comfortable atmosphere for doing my research. I would like to thank Prof. Kouichi Konno, Prof. Tadahiro Fujimoto, Prof. Taka Tanaka, associate Prof. Katsutsugu Matsuyama, and associate Prof. Naoshi Nakaya who gave me many useful advice and comments since I started my research.

I would like to thank every member of Smart Computer Vision Lab. of Iwate University for their help and patience. I would especially like to thank Dr. Chao Zhang, Dr. You Mengbo and Dr. Sun Haitian for their help, advice, and support. I would like to thank the fellow student for their help and advice.

I appreciate thanking all of my friends and previous teachers. Furthermore, I would thank my parents, who always give me unending support but never ask for any return.

Bibliography

[1] A. Fitzgibbon, Y. Wexler, and A. Zisserman, "Image-based rendering using image-based priors," *International Journal of Computer Vision*, vol. 63, no. 2, pp. 141–151, 2005.

[2] M. Aksoy, O. Torkul, and I. H. Cedimoglu, "An industrial visual inspection system that uses inductive learning," *Journal of Intelligent Manufacturing*, vol. 15, no. 4, pp. 569–574, 2004.

[3] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Computer graphics and Applications*, vol. 22, no. 2, pp. 56–65, 2002.

[4] T. Luczak and W. Szpankowski, "A suboptimal lossy data compression based on approximate pattern matching," *IEEE transactions on Information Theory*, vol. 43, no. 5, pp. 1439–1451, 1997.

[5] R. M. Dufour, E. L. Miller, and N. P. Galatsanos, "Template matching based object recognition with unknown geometric parameters," *IEEE Transactions on Image Processing*, vol. 11, no. 12, pp. 1385–1396, 2002.

[6] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in *Proceedings of the seventh IEEE international conference on computer vision*, vol. 2. IEEE, 1999, pp. 1033–1038.

[7] C.-M. Mak, C.-K. Fong, and W.-K. Cham, "Fast motion estimation for h. 264/avc in walsh–hadamard domain," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 6, pp. 735–745, 2008.

[8] Y. Moshe and H. Hel-Or, "Video block motion estimation based on gray-code kernels," *IEEE Transactions on Image Processing*, vol. 18, no. 10, pp. 2243–2254, 2009.

[9] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2. IEEE, 2005, pp. 60–65.

[10] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Transactions on image processing*, vol. 16, no. 8, pp. 2080–2095, 2007.

[11] R. Zhang, W. Ouyang, and W.-K. Cham, "Image deblocking using dual adaptive fir wiener filter in the dct transform domain," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2009, pp. 1181–1184.

[12] Y. Shin, J. S. Ju, and E. Y. Kim, "Welfare interface implementation using multiple facial features tracking for the disabled people," *Pattern Recognition Letters*, vol. 29, no. 13, pp. 1784–1796, 2008.

[13] Y. Alon, A. Ferencz, and A. Shashua, "Off-road path following using region classification and geometric projection constraints," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 1. IEEE, 2006, pp. 689–696.

[14] Q. Wang and S. You, "Real-time image matching based on multiple view kernel projection," in *2007 IEEE Conference on Computer Vision and Pattern Recognition.* IEEE, 2007, pp. 1–8.

[15] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[16] W. Ouyang, F. Tombari, S. Mattoccia, L. Di Stefano, and W.-K. Cham, "Performance evaluation of full search equivalent pattern matching algorithms," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 1, pp. 127–143, 2011.

[17] S. Oron, T. Dekel, T. Xue, W. T. Freeman, and S. Avidan, "Best-buddies similarity—robust template matching using mutual nearest neighbors," *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, vol. 40, no. 8, pp. 1799–1813, 2018.

[18] I. Talmi, R. Mechrez, and L. Zelnik-Manor, "Template matching with deformable diversity similarity," in *2017 IEEE computer society conference on computer vision and pattern recognition.* IEEE, 2017, pp. 1311–1319.

[19] X. Bai, Y. Zhang, H. Liu, and Z. Chen, "Similarity measure-based possibilistic fcm with label information for brain mri segmentation," *IEEE transactions on cybernetics*, vol. 49, no. 7, pp. 2618–2630, 2018.

[20] D. Wang, H. Lu, and C. Bo, "Visual tracking via weighted local cosine similarity," *IEEE transactions on cybernetics*, vol. 45, no. 9, pp. 1838–1850, 2014.

[21] F. Bowen, J. Hu, and E. Y. Du, "A multistage approach for image registration," *IEEE transactions on cybernetics*, vol. 46, no. 9, pp. 2119–2131, 2015.

[22] T. Dekel, S. Oron, M. Rubinstein, S. Avidan, and W. T. Freeman, "Best-buddies similarity for robust template matching," in *2015 IEEE computer society conference on computer vision and pattern recognition*, 2015, pp. 2021–2029.

[23] C. Zhang and T. Akashi, "Robust projective template matching," *IEICE TRANSACTIONS on Information and Systems*, vol. 99, no. 9, pp. 2341–2350, 2016.

[24] J. Kennedy, "Particle swarm optimization," *Encyclopedia of machine learning*, pp. 760–766, 2010.

[25] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[26] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on evolutionary computation*, vol. 11, no. 6, pp. 712–731, 2007.

[27] L. Thiele, K. Miettinen, P. J. Korhonen, and J. Molina, "A preference-based evolutionary algorithm for multi-objective optimization," *Evolutionary computation*, vol. 17, no. 3, pp. 411–436, 2009.

[28] J. Sato and T. Akashi, "Deterministic crowding introducing the distribution of population for template matching," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 13, no. 3, pp. 480–488, 2018.

[29] S.-D. Wei and S.-H. Lai, "Fast template matching based on normalized cross correlation with adaptive multilevel winner update," *IEEE Transactions on Image Processing*, vol. 17, no. 11, pp. 2227–2235, 2008.

[30] W.-H. Pan, S.-D. Wei, and S.-H. Lai, "Efficient ncc-based image matching in walsh-hadamard domain," in *European Conference on Computer Vision*. Springer, 2008, pp. 468–480.

[31] Y. Hel-Or, H. Hel-Or, and E. David, "Matching by tone mapping: Photometric invariant template matching," *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, vol. 36, no. 2, pp. 317–330, 2014.

[32] E. Elboher and M. Werman, "Asymmetric correlation: a noise robust similarity measure for template matching," *IEEE Transactions on Image Processing (TIP)*, vol. 22, no. 8, pp. 3062–3073, 2013.

[33] J.-H. Chen, C.-S. Chen, and Y.-S. Chen, "Fast algorithm for robust template matching with m-estimators," *IEEE Transactions on signal processing*, vol. 51, no. 1, pp. 230–243, 2003.

[34] O. Pele and M. Werman, "Robust real-time pattern matching using bayesian sequential hypothesis testing," *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, vol. 30, no. 8, pp. 1427–1443, 2008.

[35] H. Y. Kim and S. A. De Araújo, "Grayscale template-matching invariant to rotation, scale, translation, brightness and contrast," in *Pacific-Rim Symposium on Image and Video Technology (PSIVT)*. Springer, 2007, pp. 100–113.

[36] A. Penate-Sanchez, L. Porzi, and F. Moreno-Noguer, "Matchability prediction for full-search template matching algorithms," in *2015 International Conference on 3D Vision*. IEEE, 2015, pp. 353–361.

[37] S. Korman, D. Reichman, G. Tsur, and S. Avidan, "Fast-match: Fast affine template matching," in *2013 IEEE computer society conference on computer vision and pattern recognition*, 2013, pp. 2331–2338.

[38] D.-M. Tsai and C.-H. Chiang, "Rotation-invariant pattern matching using wavelet decomposition," *Pattern Recognition Letters*, vol. 23, no. 1-3, pp. 191–201, 2002.

[39] C. Zhang and T. Akashi, "Fast affine template matching over galois field," in *British Machine Vision Conference*. BMVA Press, September 2015, pp. 121.1–121.11.

[40] B. G. Shin, S.-Y. Park, and J. J. Lee, "Fast and robust template matching algorithm in noisy image," in *Control, Automation and Systems (ICCAS)*. IEEE, 2007, pp. 6–9.

[41] A. Sibiryakov, "Fast and high-performance template matching method," in *2011 IEEE computer society conference on computer vision and pattern recognition*. IEEE, 2011, pp. 1417–1424.

[42] W. Ouyang, F. Tombari, S. Mattoccia, L. Di Stefano, and W.-K. Cham, "Performance evaluation of full search equivalent pattern matching algorithms," *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, vol. 34, no. 1, pp. 127–143, 2012.

[43] J. Hafner, H. S. Sawhney, W. Equitz, M. Flickner, and W. Niblack, "Efficient color histogram indexing for quadratic form distance functions," *IEEE transactions on pattern analysis and machine intelligence*, vol. 17, no. 7, pp. 729–736, 1995.

[44] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *2000 IEEE computer society conference on computer vision and pattern recognition*. IEEE, 2000, pp. 142–149.

[45] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *European Conference on Computer Vision (ECCV)*. Springer, 2002, pp. 661–675.

[46] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, "Comparing images using the hausdorff distance," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 15, no. 9, pp. 850–863, 1993.

[47] M.-P. Dubuisson and A. K. Jain, "A modified hausdorff distance for object matching," in *Proceedings of 12th international conference on pattern recognition.* IEEE, 1994, pp. 566–568.

[48] O. Jesorsky, K. J. Kirchberg, and R. W. Frischholz, "Robust face detection using the hausdorff distance," in *International conference on audio-and video-based biometric person authentication.* Springer, 2001, pp. 90–95.

[49] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International journal of computer vision (IJCV)*, vol. 40, no. 2, pp. 99–121, 2000.

[50] S. Oron, A. Bar-Hillel, D. Levi, and S. Avidan, "Locally orderless tracking," *International Journal of Computer Vision (IJCV)*, vol. 111, no. 2, pp. 213–228, 2015.

[51] D. Simakov, Y. Caspi, E. Shechtman, and M. Irani, "Summarizing visual data using bidirectional similarity," in *2008 IEEE computer society conference on computer vision and pattern recognition.* IEEE, 2008, pp. 1–8.

[52] T.-t. Li, B. Jiang, Z.-z. Tu, B. Luo, and J. Tang, "Image matching using mutual k-nearest neighbor graph," in *International Conference of Young Computer Scientists, Engineers and Educators.* Springer, 2015, pp. 276–283.

[53] K. Ozaki, M. Shimbo, M. Komachi, and Y. Matsumoto, "Using the mutual k-nearest neighbor graphs for semi-supervised classification of natural language data," in *Proceedings of the fifteenth conference on computational natural lan-*

guage learning. Association for Computational Linguistics, 2011, pp. 154–162.

[54] H. Liu, S. Zhang, J. Zhao, X. Zhao, and Y. Mo, "A new classification algorithm using mutual nearest neighbors," in *2010 Ninth International Conference on Grid and Cloud Computing*. IEEE, 2010, pp. 52–57.

[55] Z. Hu and R. Bhatnagar, "Clustering algorithm based on mutual k-nearest neighbor relationships," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 5, no. 2, pp. 100–113, 2012.

[56] D. E. Goldberg and J. H. Holland, "Genetic algorithms and machine learning," 1988.

[57] L. Davis, "Handbook of genetic algorithms," 1991.

[58] Z. Michalewicz and M. Schoenauer, "Evolutionary algorithms for constrained parameter optimization problems," *Evolutionary computation*, vol. 4, no. 1, pp. 1–32, 1996.

[59] U. Maulik and S. Bandyopadhyay, "Genetic algorithm-based clustering technique," *Pattern recognition*, vol. 33, no. 9, pp. 1455–1465, 2000.

[60] W.-B. Tao, J.-W. Tian, and J. Liu, "Image segmentation by three-level thresholding based on maximum fuzzy entropy and genetic algorithm," *Pattern Recognition Letters*, vol. 24, no. 16, pp. 3069–3078, 2003.

[61] H. Vafaie and K. A. De Jong, "Genetic algorithms as a tool for feature selection in machine learning." in *ICTAI*, 1992, pp. 200–203.

[62] D. Whitley *et al.*, "Genetic algorithms and neural networks," *Genetic algorithms in engineering and computer science*, vol. 3, pp. 203–216, 1995.

[63] N. Senthilkumaran and R. Rajesh, "Edge detection techniques for image segmentation–a survey of soft computing approaches; ijrte, vol. 1, no. 2, may 2009," 2009.

[64] V. Ayala-Ramirez, C. H. Garcia-Capulin, A. Perez-Garcia, and R. E. Sanchez-Yanez, "Circle detection on images using genetic algorithms," *Pattern Recognition Letters*, vol. 27, no. 6, pp. 652–657, 2006.

[65] J. Dehmeshki, X. Ye, X. Lin, M. Valdivieso, and H. Amin, "Automated detection of lung nodules in ct images using shape-based genetic algorithm," *Computerized Medical Imaging and Graphics*, vol. 31, no. 6, pp. 408–417, 2007.

[66] S. K. Kim, D. W. Kim, and H. J. Kim, "A recognition of vehicle license plate using a genetic algorithm based segmentation," in *Image Processing, 1996. Proceedings., International Conference on*, vol. 2. IEEE, 1996, pp. 661–664.

[67] Y. Zhang, C. Zhang, and T. Akashi, "Localization of vehicle inspection sticker in a single image," in *Fourth International Symposium on Future Active Safety Technology Toward Zero Traffic Accidents*, 2017, pp. TuB–P2–4.

[68] Y. zhang, C. zhang, and T. Akashi, "Localization of vehicle inspection sticker with projective transformation and constraints," in *The 24th Symposium on Sensing via Image Infomation (SSII2017)*, 2017, pp. IS3–08.

[69] Y. Zhang, C. Zhang, and T. Akashi, "Multi-scale template matching with scalable diversity similarity in an unconstrained environment," in *British Machine Vision Conference*. BMVA, 2019, pp. 222.1–222.11.

[70] H. Farid, "Blind inverse gamma correction," *IEEE transactions on image processing*, vol. 10, no. 10, pp. 1428–1433, 2001.

[71] E. Reinhard, W. Heidrich, P. Debevec, S. Pattanaik, G. Ward, and K. Myszkowski, *High dynamic range imaging: acquisition, display, and image-based lighting*. Morgan Kaufmann, 2010.

[72] P. Babakhani and P. Zarei, "Automatic gamma correction based on average of brightness," *Advances in Computer Science: an International Journal*, vol. 4, no. 6, pp. 156–159, 2015.

[73] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition*. IEEE, 2005, pp. 886–893.

[74] Y. Zhang, M. You, T. Miyoshi, and T. Akashi, "A vision system for canning with fish sensing using rule-based matching and segmentation," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 15, no. 6, pp. 956–964, 2020.

[75] I. C. Trelea, "The particle swarm optimization algorithm: convergence analysis and parameter selection," *Information processing letters*, vol. 85, no. 6, pp. 317–325, 2003.