Recognition of Planar Object in Stack via Local Prediction Consistency

Jianying Bao, Lei Feng, Chengge Gu, Jinqiu Mo
School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
agnes_0130@sjtu.edu.cn

Abstract. A novel method for recognition of planar object in stack via local prediction consistency is proposed in this paper. By checking the consistency of local predictions, an object can be recognized and located. The prediction method by local features under uncertain scale is emphatically discussed. According to the distribution prior for local predictions, the similarity scale can be determined and the consistent local predictions can be composited. A great number of experiments are carried out to prove the stability of this method.

1. Introduction
In industry application, the information obtained in recognition of workpieces in stack can contribute to following tasks such as bin-picking and automatic assembling. Traditional industrial methods of workpiece sorting commonly adopt mechanical feeding and discharge structures which are specially designed according to the shape features of different workpieces in the conveyer belt. In recent years, some novel approaches have been applied to locating workpieces in stack. Among them, combining machine vision with the industrial robot is a popular research direction. Machine vision can be used to extract environmental information needed for the industrial robot. The novel methods combining machine vision with the industrial robot can make the production line flexible, cost-saving, space-saving and easy-maintainable.

The method proposed in this paper generates pose information of planar thin objects in stack via local prediction consistency. Compared to traditional methods, our method has advantages of occupying less memory as well as high recognition speed, and in certain condition the searching in scale space can be omitted. Moreover, this method is accurate and robust to occlusion and clutter, and it has extensive application prospect in industrial tasks about planar workpieces such as automatic grasp for the planar sheet metal.

1.1. Related work
Recognizing and locating planar objects with a 2D camera has been researched for many years and there are lots of industrial applications. A recognition method for planar objects with monocular camera is proposed in [1], where a modified Generalized Hough Transform voting scheme based on edge points is employed. Then the edgelet is used as the local descriptor, CAD models are discretized and accumulators are used to store the value of translation, scale, and rotation around the x, y and z axes. Finally, an optimization framework is built for object pose refinement and the first match success can reach up to 97% for some experimented objects. In [2], right and left stereo views are applied to the system to estimate poses of objects. And landmark features are chosen to simplify the recognition. The
proposed algorithm is used to find corresponding features in the right and left stereo views and then extract the useful information to refine poses of objects. It is verified to perform well in recognition tasks about objects in stack with simple landmark features. In [3], the photometric stereo method is used to generate the surface normal distribution of the scene. And poses of objects are determined by comparing the distribution with the extended-Gaussian-image. This method can be applied to objects with obvious characters in normal direction of the surface like donut and it has no demand for the exact size of objects. In [4], a multi-flash camera is used to extract robust depth edges and then the chamfer matching method is used to detect objects and estimate their poses as a result to largely improve the matching accuracy. Moreover, in order to improve the accuracy, a two-view approach is adopted to refine the result after grasping objects. In [5], a RGB-D camera is used to get the scene point cloud during depalletizing. By pre-segmentation the point cloud of each object is separated and then the pose of the object can be estimated precisely through expectation-maximization (EM) algorithm. This approach is applicable for the well-separated objects rather than tightly stacked objects.

2. Object Detection

The objects in stack should be accurately located. However, an object whose information is incomplete due to occlusion in the image is too difficult to be recognized, so local features are used to estimate object pose, for they are more likely to be found from the incomplete object information. This process is consistent with the recognition way of human. A method to recognize the object via local prediction consistency (LPC) is proposed in this paper. The operation process of LPC is shown in Fig.1. First, local features are selected. Then, according to the matching results of local features and the relative position of each local feature in the whole object, the prediction of the object’s pose can be generated. Finally, examination for the consistency among different local feature predictions should be performed. On the basis of reliability analysis in [6], more consistent local predictions can be considered as a more reliable existent estimate of the object. And the low consistency of local prediction can be considered as misrecognition to be rejected. In general, the probability of the prediction about the existence of the object is almost one when the prediction contains at least three consistent local predictions.

![Fig.1 Operational process of LPC](image)

2.1. Selection of local feature

Local features are representative contour segments of the object. As long as the contour segment matching result is not ambiguous, it is more appropriate to choose the small contour segment because it is less likely to be occluded in the scene. That is, there will be more local feature instances to be detected and high local prediction consistency results are more likely to be obtained.

Considering the sensitivity to scale, local feature can be divided into two categories, one is sensitive to similarity transformation and the other is not. In image matching, a fixed scale will be got by the scale-sensitive feature. However, as to the scale-insensitive feature, it can be well matched under any scale, so the exact scale of a whole object cannot be determined by the prediction of scale-insensitive features.

An object may have some repeating local features, for example if two local features are both corners and with same angle, then their matching results will have no difference. In order to avoid such repeating matching, only one template of repeating features is used for the instance detection and its detection results are shared by the other repeating features.
See the object in Fig.2. The features in red boxes are selected as local features. Features in box 2 and box 6 are scale-sensitive features, and features in box 1 and box 4 are scale-insensitive features. And features in box 1 and box 3 are repeating features.

![Fig.2 Object for recognition](image)

2.2. Candidate extraction

As to scale-sensitive features, according to the relative position of local features in the whole object, the hypothesis of the object’s existence can be obtained. The pose of the object can be obtained with two parameters: \( \Delta \theta \) and \( V_t \):

\[
P_{om}^{(i)} = p_{os}^{(i)} + T(\theta_s^{(i)}) \cdot V_t
\]

\[
\theta_m^{(i)} = \theta_s^{(i)} + \Delta \theta
\]

Where, \( \Delta \theta \) is the difference between \( V_m \) and \( V_s \) shown in Fig.3. Here for scale-sensitive features, the position prediction is a point and we name it point prediction.

![Fig.3 Local-global prediction method without scaling](image)

As to scale-insensitive features, a local-global prediction method that takes scale into account is performed. Take a corner feature of a particular angle for an example. Prediction results under different scales can be obtained from the result under a certain scale through the scaling fixed point. The matching method for scale-insensitive features is illustrated in Fig.4.

![Fig.4 Match of sub-template under different scales](image)

As is shown in Fig.4, let the local template of detection instance be 1, the reference point be \( P_o \) and corner point be \( C \). The relative position between the corner point and the reference point in the standard template is described by the vector \( V_{et} = V_{et}(1) = \overrightarrow{P_oC} \), then the relative position in the template under scale of \( s \) will be:

\[
V_{et}(s) = s \cdot V_{et} = s \cdot \overrightarrow{P_oC}
\]

Therefore,

\[
P_{os1} = p_o + \overrightarrow{P_oC} + \overrightarrow{C_{os1}} = p_o + V_{et} - V_{et}(s) = p_o + V_{et}(1-s)
\]

If the orientation of the i-th detection instance is \( \theta_s^{(i)} \), then the relative position between its corner point and its reference point is:
\[
V_t \left(1, \theta_s^{(i)}\right) = T \left(\theta_s^{(i)}\right) \cdot \bar{P}_o \bar{C}
\]

(5)

Correspondingly, the position of the instance’s center point under scale of \(s\) is:

\[
p_{oxt}^{(i)} = p_o^{(i)} + \left(1 - s\right) \cdot V_t \left(1, \theta_s^{(i)}\right) = p_o^{(i)} + V_t \left(1 - s, \theta_s^{(i)}\right)
\]

(6)

According to (1) and (2), the pose prediction of the whole object can be obtained under scale of \(s\):

\[
p_m^{(i)} = p_{oxt}^{(i)} + T \left(\theta_s^{(i)}\right) \cdot s \cdot V_t
\]

(7)

\[
\theta_m^{(i)} = \theta_s^{(i)} + \Delta \theta
\]

(8)

Combining (6), (7) and (8):

\[
p_m^{(i)} = s \cdot T \left(\theta_s^{(i)}\right) \left(V_t - P_o \bar{C}\right) + P_o^{(i)} + T \left(\theta_s^{(i)}\right) \cdot P_o \bar{C}
\]

(9)

(9) shows that, the position prediction is a ray with a starting point \(P_o^{(i)} + T \left(\theta_s^{(i)}\right) \cdot P_o \bar{C}\) and a direction \(T \left(\theta_s^{(i)}\right) \left(V_t - P_o \bar{C}\right)\). Generally, it is necessary to limit the range of scale in practice, so that for scale-insensitive features, the position prediction would be a line segment and we name it line prediction.

After obtaining predictions of multiple local features, consistent predictions should be sorted out. Here we define \(n\)-order prediction as the prediction that contains \(n\) consistent local feature predictions. If an object has \(m\) local features, this object can have \(m\)-order prediction at most. From two-order prediction to \(m\)-order prediction, each order should be calculated. According to the consistency measure formula in [6]:

\[
P \left(H_{obj}(f_i)^{(p)} = H_{obj}(f_j)^{(q)}\right) = P \left(position_i^{(p)} = position_j^{(q)}\right) \cdot P \left(angle_i^{(p)} = angle_j^{(q)}\right)
\]

(10)

Where:

\[
P \left(position_i^{(p)} = position_j^{(q)}\right) = \exp \left(- \frac{\|position_i^{(p)} - position_j^{(q)}\|^2}{\beta_p}\right)
\]

(11)

\[
P \left(angle_i^{(p)} = angle_j^{(q)}\right) = \exp \left(- \frac{\|angle_i^{(p)} - angle_j^{(q)}\|^2}{\beta_a}\right)
\]

(12)

Therefore, the consistency under two-order prediction can be calculated directly. If one is a line prediction and the other is a point prediction, the line should be converted into a point under the scale of point prediction. And if both are lines, the maximum consistency under different scales will be chosen.

The consistency under \(n\)-order prediction can be calculated from \((n-1)\)-order prediction. Symbol brace \{\} is used to represent a feature sequence. For example, \(seq_3 = \{1,2,3\}\) is a sequence containing three features, and each number represent a corresponding local feature. Considering one \(n\)-order prediction of an object with \(m\) local features, the sequence is denoted by:

\[
seq_m^{(i)} = \{a_1, a_2, a_3, ..., a_n\}, a_i \in \{1,2,3, ..., m\}\] and \(a_i < a_j, for i < j\)

The sequence it contains under \((n-1)\)-order prediction is denoted by \(seq_{n-1}^{(i)} = \{a_1, a_2, a_3, ..., a_{i-1}, a_{i+1}, ..., a_n\}\) \((1 \leq l \leq n)\), and then the \(n\)-order prediction can be obtained by:

\[
\{a_1, ..., a_n\} = \{a_1, ..., a_{i-1}, a_{i+1}, ..., a_n\} \land \{a_1, a_i\} \land \{a_2, a_i\} \land ... \land \{a_i, a_n\}
\]

(13)

The symbol \(\land\) means to combine the features of prediction sets. The operation defined by this symbol should obey commutative law and associative law, and a large sequence can absorb the smaller one it contains.

With \(l\) ranging from 1 to \(n\), \(n\) different equations can be obtained from (13). Then, use operator \(\land\) to combine the corresponding sides of all these equations:

\[
\{a_1, ..., a_n\} = \bigwedge_{l=1}^{n} \{a_1, ..., a_{l-1}, a_{l+1}, ..., a_n\}
\]

(14)

If \(comb(n,s)\) is used to represent the sequence set for choosing \(n\) features to be a sequence for \(s\) features in total, the sequence set under any order can be written as:

\[
\{a_1, ..., a_n\} = \bigwedge_{s=2}^{n} comb(s, \{a_1, ..., a_n\}), for 2 \leq s < n
\]

(15)

To sum up, the strategy of obtaining multi-order predictions can be performed as follow:

1) Calculate all two-order prediction sets.
2) Obtain three-order prediction sets by \(seq_3 = \bigwedge \{comb(2, \{a_1, ..., a_n\})\} \).

4
3) Obtain the following prediction sets: \( s_{eq_{i+1}} = \text{\text{comb}}(i, \{a_1, ..., a_n\}) \), for \( i = 3 \sim n-1 \).

After obtaining all prediction sets, the consistent predictions in different set should be synthesized. If there exist scale insensitive features in some set, the exact scale should be determined.

Suppose that there is a \( n \)-order prediction set, and in this set there are \( p \) sensitive features and \( q \) scale insensitive features. The scale range is \( \text{[scaleMin, scaleMin]} \). The point prediction is denoted by:

\[
\text{Pred}_{\text{point}} = \{P_i = (x_i, y_i)^T, \text{angle}_i, \text{scale} = s_i, i = 1, 2, 3 \ldots p\}
\]

Where \( P_i \) and \( \text{angle}_i \) represent each pose position and orientation prediction of the object according to the corresponding feature, and \( s_i \) represents each matching scale.

Likewise, the line prediction is denoted by:

\[
\text{Pred}_{\text{line}} = \{L_j = (x_j, y_j)^T, d_j = (a_j, b_j)^T, \text{angle}_j, j = 1, 2, 3 \ldots q\}
\]

Where \( \text{angle}_j \) represents each orientation prediction of the object, \( L_j \) represents each position prediction of the object under scale of one, and \( d_j \) represents direction vector per unit scale of line-prediction. Here the orientation prediction will not change with scale, so in the following discussion, only the position prediction is considered.

The true pose prediction must be in close proximity to the real pose, so two targets for determining result scale can be given:

1. The position predictions must be close enough to each other in space;
2. The difference between the result scale and each matching scale of point prediction should be as small as possible;

The first target can be quantified by the sum of distances between each prediction and their mean, written by target function:

\[
\min f(s) = \sum_{i=1}^{p} \|P_i - P_{\text{avg}}\|_2^2 + \sum_{j=1}^{q} \|L_j + (s - 1) \cdot d_j - P_{\text{avg}}\|_2^2
\]

Where \( P_{\text{avg}} \) is the mean and it is obtained by:

\[
P_{\text{avg}} = \frac{1}{n} \left( \sum_{i=1}^{p} P_i + \sum_{j=1}^{q} (L_j + (s - 1) \cdot d_j) \right) = \frac{1}{n} \left( s \cdot \sum_{j=1}^{q} d_j + \sum_{j=1}^{q} (L_j - d_i) + \sum_{i=1}^{p} P_i \right)
\]

Similarly, the second target can be quantified by:

\[
\min g(s) = \sum_{i=1}^{p} (s - s_i)^2 = p \cdot s^2 - 2s \sum_{i=1}^{p} s_i + \sum_{i=1}^{p} s_i^2
\]

Considering both two targets at the same time, the joint optimization objective function can be expressed as:

\[
\min F(s) = \min \left( f(s) + \lambda \cdot g(s) \right)
\]

Where \( \lambda \) represents the relative weight between two targets in joint optimization objective function.

According to the first order necessary condition for optimization \( \frac{dF(s)}{ds} = 0 \), we have:

\[
s = -\frac{-nH^T(B - H + A) + r - h - \lambda u}{(h - nH^TH + \lambda p)}
\]

If the \( s \) is out of scale range, choose the nearer marginal value to \( s \) as the result scale.

So here exact \( s \) is obtained. And each line-prediction can be converted to point-prediction by this exact scale. And then the final prediction result can be synthesized by weighted average method and the like in [6].

3. Experiments
Our test system is built with a color RGB camera whose resolution is 2592*1944. The illumination system of this platform consists of four natural lights whose brightness is adjustable round the base of this platform. The software is implemented in Matlab with the Halcon library for image processing. In consistency measure formulas, \( \beta_a \) is set to 38 and \( \beta_s \) is set to 5932. The threshold of the consistency measure formula for two objects is set to 0.8, and the lowest prediction’s order to accept is set to 5, i.e.,
only the prediction with consistency order higher than five is accepted as reliable result.

3.1. Preprocessing
The Steger matching method[7] is used to implement the matching step for local feature instances. Therefore, in the preprocess for noise smoothing, the edge information should be reserved as much as possible. We choose the mean curvature flow operator in Halcon for image preprocessing. This operator can smooth the noise, meanwhile the object edges can be reserved. The original image is shown in Fig.5(a). Fig.5(b) is the edge extraction result from original image in Fig.5(a) while the result in Fig.5(b) is from original figure processed by mean curvature flow operator. In the mean curvature flow operator, both sigma and theta are set to 0.5, and the number of iterations is set to 9.

![Fig.5 Extracted edge before and after preprocessing](image)

3.2. Experiment result
We test over 100 scenes with objects in stack. The final experiment result is shown in Table 1. With the same scenes, cross correlation method and the overall shape matching method in Halcon are also tested and both thresholds are set to 0.6. The results are also shown in Table 1.

The results show that accuracy of LPC described in this paper for objects recognition can reach up to 99%, and it is much higher than other two methods. Therefore, LPC is proved to have a strong stability. Some images of results are shown in Fig.7. Left images are original images and right images are recognition results by LPC.

![Fig.6 Experiment objects](image)

| Experimental object | Object 1 | Object 2 |
|---------------------|----------|----------|
| Item                | Number of experiments | Match accuracy | Number of experiments | Match accuracy |
| LPC                 | 100      | 100%     | 105      | 99%      |
| Overall shape matching | 100    | 78%      | 105      | 90%      |
| Cross correlation method | 100    | 84%      | 105      | 70%      |
4. Conclusion and Future Works

A novel method for recognition and location of planar objects in stack via local prediction consistency is proposed in this paper. The prediction method by local features under uncertain scale is emphatically discussed. The similarity scale is determined and consistent local predictions is composited on the basis of the distribution prior for local predictions. According to the experimental results, this method is proved to be robust to occlusion.

Future work will focus on the improvement in the algorithm in order to speed up the recognition process. And the recognition of objects under affine transformation will be researched.

References:
[1] Alberto Pretto, Stefano Tonello, et al. Flexible 3D localization of planar objects for industrial bin-picking with monocamera vision system[P]. IEEE International Conference on Automation Science and Engineering (CASE), 2013:168-175.
[2] Rahardja, Krisnawan & Kosaka, et al. Vision-based bin-picking: recognition and localization of multiple complex objects using simple visual cues[C]. IEEE International Conference on Intelligent Robots and Systems, 1996:1448-1457.
[3] Katsushi Ikeuchi, Berthold K.P. Horn, et al. Picking Up an Object from a Pile of Objects[C]. 1st ISRR, 1983.
[4] Liu M Y, Tuzel O, et al. Fast object localization and pose estimation in heavy clutter for robotic bin-picking[J]. The International Journal of Robotics Research, 2012, 31(8):951-973.
[5] Dirk Holz, Angeliki Topalidou-Kyiazopoulou, et al. Real-time object detection, localization and verification for fast robotic depalletizing[C]. IEEE International Conference on Intelligent Robots & Systems, 2015:1459-1466.
[6] GU Chengge, BAO Jianying, et al. Planar object recognition based on local feature prediction agreements[C]. MATEC Web of Conferences, 2018(in Chinese).
[7] Steger C. Similarity Measures for Occlusion, Clutter, and Illumination Invariant Object Recognition[M]. Pattern recognition :23rd DAGM Symposium, 2001:148-154.