Contour Primitive of Interest Extraction Network Based on One-shot Learning for Object-Agnostic Vision Measurement

Fangbo Qin, Jie Qin, Siyu Huang, De Xu*, Senior Member, IEEE

Abstract—Image contour based vision measurement is widely applied in robot manipulation and industrial automation. It is appealing to realize object-agnostic vision system, which can be conveniently reused for various types of objects. We propose the contour primitive of interest extraction network (CPieNet) based on the one-shot learning framework. First, CPieNet is featured by that its contour primitive of interest (CPI) output, a designated regular contour part lying on a specified object, provides the essential geometric information for vision measurement. Second, CPieNet has the one-shot learning ability, utilizing a support sample to assist the perception of the novel object. To realize lower-cost training, we generate support-query sample pairs from unpaired online public images, which cover a wide range of object categories. To obtain single-pixel wide contour for precise measurement, the Gabor-filters based non-maximum suppression is designed to thin the raw contour. For the novel CPI extraction task, we built the Object Contour Primitives dataset using online public images, and the Robotic Object Contour Measurement dataset using a camera mounted on a robot. The effectiveness of the proposed methods is validated by a series of experiments.

Index Terms—Deep learning for visual perception, computer vision for automation, object detection, segmentation and categorization.

I. INTRODUCTION

Vision measurement means using camera to precisely sense the spatial pose and structure of a viewed object, which is widely applied in robotic tasks and industrial automation [1-3]. Compared to other computer vision tasks, vision measurement focuses on geometric representation and spatial pose, instead of textured appearance, dense reconstruction, and category identification. Besides, for many robotic and industrial cases requiring high precision, the coarse visual perception is not sufficient. Therefore, geometric image feature extraction is an essential issue for vision measurement. Combining extracted image features and calibrated imaging model, 3D spatial information can be calculated [4,5].

Object contours are widely used in vision measurement.

First, contour feature is more robust to partial occlusion and missing than point feature. Second, contour provides the sparse and informative geometric representation of object. In [6], the circle contours were used to measure the 3D position of drogue of aerial vehicle. In [7], the pose measurement of space non-cooperative target was realized based on both circle and line contour parts [7]. For the pose alignment of high precision devices, a set of line segments on the objects’ end-faces was used to reflect the spatial pose error [8]. In these works, the feature extraction methods were only suitable for the specified object types, and not scalable for novel objects.

To improve the intelligence and scalability of image based vision measurement, it is appealing to address the following two problems: object-agnostic vision measurement and contour of interest extraction. First, inspired by the recent works on class-agnostic vision [9-11], we attempt to explore the object-agnostic geometric image feature extraction, so that a vision measurement system can be flexibly applied to various objects over different scenarios. Second, instead of extracting general contours globally, a measurement task mainly concerns a set of contours of interest, which are highly related to the task purpose and have geometric meaning.

Towards precise object-agnostic vision measurement with better reusability and scalability, this paper aims to realize the end-to-end object-agnostic contour of interest extraction. Our contribution is as follows:

1) The contour primitive of interest extraction network (CPieNet) is proposed based on one-shot learning, which extracts a set of pixels representing a specified contour primitive of interest (CPI) on an object from its raw image. One-shot learning enables the model to work on novel object by involving a support sample with annotation.

2) To obtain the one-pixel CPI, a Gabor-filters based non-maximum suppression (GF-NMS) method is proposed to thin the raw CPI output by CPieNet.

3) Because it is costly and tedious to capture and annotate numerous support-query image pairs of objects, we design an automatic sample pair generation method, which converts a single annotated online public image to a support-query sample pair by random transformation.

4) To the best of our knowledge, this work is the first to explore the one-shot learning of CPI extraction. For this novel task, we built the Object Contour Primitives (OCP) dataset using online public images, and built the Robotic Object Contour Measure (ROCM) dataset using images of 15 objects collected by an eye-in-hand industrial robot.
II. RELATED WORKS

A. Object-agnostic Contour Based Vision Measurement

Image contour based visual measurement is preferred due to its guaranteed accuracy, robustness and sparsity. In recent years, several systems that can be reused among different object types were developed. He et al. proposed a sparse template based 6D pose estimation method for industrial metal parts, which relies on line segment detection and cannot work on circular-shape objects [12]. In [13], the silhouette contour was extracted and used to match the nearest template, for pose estimation of textureless object, whose real-time performance was limited. In [14], the contour primitives of interest extraction (CPIE) method was proposed, which used a CPI template to match the object, then executed pixel-level analysis near the matched CPIs for precise geometric calculation. CPIE is only effective in high precision vision with grayscale image and highly structured scene. In comparison, CPieNet is fast and end-to-end, inferring CPIs directly from raw image. Besides, deep learning technology brings the promising generalization ability over various objects and conditions.

B. Deep Learning Based Contour Detection

Deep learning based contour detection models outperform traditional methods, due to its powerful hierarchical feature learning ability [15]. Edge detection and boundary prediction are similar tasks [16,17]. Semantic edge detection not only extract the edge pixels but also tells which category of object each edge belongs to [18]. Line segment detection parses the line-like contours [19]. These methods presented the promising performance of deep learning on contour-related perception, but provided general low-level features which lack the task-awareness. CPieNet focuses on CPIs that have geometric meaning and are of interest to a measurement task.

C. One/few-shot Learning for Image Perception

One/few-shot learning aims to overcome the data scarcity problem in deep learning. Especially in robotic and industrial applications, it is impracticable to build a training dataset every time a novel-type object is given. According to the recent methods, given a query sample containing a novel object, its perception can be helped by one or a few annotated support images of the same object, providing a prototype vector describing the object based on masked average pooling (MAP). PANet densely compared the query image’s feature map with the prototype using cosine distance as metric, and the prototype alignment regularization (PAR) was used in training [20]. In [21], feature weighting was applied before dense comparison to encourage the higher feature response of foreground. CANet alternately used concatenation instead of cosine distance for dense comparison, and the iterative optimization module was designed to refine the result [10]. SG-One realized the one-shot similarity guidance, using the cosine similarity between prototype and guidance features to reweight the features in segmentation branch [22]. Without using MAP, A-MCG used the foreground in raw image to provide guidance [23], which is not suitable for CPI extraction because CPI foreground cannot provide object-related contextual features. Although CPI extraction can be seen as a variant of one-shot semantic segmentation, the difference between the regular-shaped narrow CPIs and the arbitrary-shaped blocky segmentation regions causes that the above existed methods are not ideally appropriate to CPI extraction task.

III. PROBLEM DEFINITION

Contour primitive (CP) is a regular contour segment, ignoring the irregular and fragmentized contour parts. As illustrated in Fig. 1, the metal part’s contour is mainly composed by several line segments and a circle. Further, CPI means a designated CP on the target object, for example, “the right long side of the aluminum part” in Fig. 1(a). Viewing a wide range of object categories, we conclude that a majority of industrial and daily objects have the two typical CPs: line segment (LS) and circular arc (CA). In addition, LS and CA are easy for shape fitting and suitable for geometric calculation.

An RGB image of the object to measure is captured and regarded as the query image \( I_0 \). The task is to extract one of the CPIs on this object based on the one-shot learning CNN model \( F \). Assuming the object type is novel and unseen during model training, a support image \( I_S \) of the same object and its CPI annotation \( C_5 \) are used to guide the query image’s perception. \( C_5 \in \{0,1\} \) is represented by a binary map, whose foreground pixels mark the CPI. Thus, the CNN model is expected to extract the corresponding CPI \( C_0 \) from \( I_0 \), namely,

\[
C_0 = F(I_0; I_S, C_5)
\]  

The task difficulty is influenced by the difference between \( I_S \) and \( I_0 \). In the current work, we assume that no repeated objects occur, and the variation of imaging condition is controlled, including translation, limited rotation, illumination change, color change, background change, and other stuff’s occurrence. The large view-angle changes and cluttered scene are not involved. Fortunately, in many robotic manipulation and industrial applications, the viewed scenes are usually controlled. And the coarse visual perception techniques can be leveraged to control the view point, region of interest, etc. Therefore, with the controlled difference between \( I_S \) and \( I_0 \), it is feasible and meaningful to realize precise CPI extraction.

Because vision measurement usually requires multiple CPIs for geometric calculation, the single CPI extraction mode described by (1) can be easily extended to the batch of CPIs extraction mode, based on GPU’s parallel computation ability,

\[
C_0^k = F(I_0^k; I_S, C_5)
\]  

where \( k=1,2,...,N_{CPI} \). Thus, given \( N_{CPI} \) query images, the predicted CPI maps are inferred in parallel, with the shared \( I_S \), \( I_0 \) and \( F \).
IV. METHODS

A. Model Architecture

CPieNet utilizes a support branch to guide the query branch. The support branch gains the prototype vector \( P \) from the support image \( I_S \). As shown in Fig. 2, \( I_S \) is fed to the backbone implemented by ResNet-50 [24], and the output size is 1/16 of the input size. An atrous spatial pyramid pooling (ASPP) module [25] is used to enlarge the spatial receptive field, whose depth is 128 and atrous rates are \( \{2, 4, 8, 16\} \). The original concatenation based fusion in ASPP is replaced by the sum based fusion. The three deepest feature maps, whose sizes equal 1/2, 1/4, and 1/8 of the input size, are drawn out from backbone, then adapted to 16, 32, and 64 channels using three 1x1 convolutional layers with 1x1 stride, respectively. These three adapted feature maps and the ASPP output are combined to form the 240-channel multi-scale representation, which are resized to 1/2 of the inputs size using bilinear interpolation, and concatenated as \( H_{50} \). Note that batch normalization is used after each of the above 1x1 convolutional layers as well as the ASPP’s last convolutional layer, to normalize the features at different scales before fusion.

The two 3x3 convolutional layers with 128 filters and 1x1 stride are used to fuse the multi-scale features in \( H_{50} \), and the resulting support feature map is \( H_S \). Note that ReLU activation is not used in the second convolutional layer. The query branch shares the same backbone and convolutional layers with the support branch. Sharing the same weights of backbone and convolutional layers with the support branch, the query branch gains the multi-scale feature map \( H_{Q0} \) and the query feature map \( H_{Q} \) from the query image \( I_Q \).

With the support feature map \( H_S \) and the annotated binary map \( C_S \), the 128-channel prototype vector \( P \) representing the CPI is obtained by masked average pooling,

\[
P = \frac{\sum_{i,j,k} H_{S(i,j,k)} \times C_{S(i,j,k)}}{\sum_{i,j,k} C_{S(i,j,k)}}
\]

where \((i,j)\) and \(k\) indicate the indices of pixel position and feature channel, respectively.

The main guidance from support branch to query branch is based on the distance measure between \( P \) and the pixels on \( H_Q \). The cosine distance is measured by,

\[
D_{ij} = \alpha \left( \frac{H_{Q(i,j)} \cdot P}{\|H_{Q(i,j)}\|_2 \times \|P\|_2} \right)
\]

where \( \alpha \) is a scaling factor. Thus, \( D_{ij} \) ranges from zero to \( 2\alpha \). Similar to [20], we set \( \alpha = 20 \) empirically. Euclidean distance can also be used for distance measure,

\[
D_{ij} = \sqrt{\sum_{k} \left( H_{Q(i,j,k)} - P_k \right)^2}
\]

After distance measure, the distance map \( D \) is fed to an output activation layer to obtain the query CPI map \( C_Q \), realized using the sigmoid function,

\[
C_{Q(i,j)} = \frac{1}{1 + e^{-\beta(D_{ij}-\beta)}}
\]

where \( \beta \) is a bias, which is set to \( \beta = 5 \), so that the activation value \( C_{Q(i,j)} \) approaches 1 when \( D_{ij} \) approaches zero. When \( D_{ij} \) is larger than 10, \( C_{Q(i,j)} \) is approximately zero.

Intuitively, the distance measure and sigmoid activation realize that the pixel has high output response if and only if its feature vector is close enough to the prototype vector. Finally, \( C_Q \) is resized back to the input size by \( 2\times \) upsampling, and the default thresholding procedure sets \( C_{Q(i,j)} \) to zero if \( C_{Q(i,j)} < 0.5 \). The raw map \( C_Q \) is further processed by the GF-NMS module to thin the contour, as introduced in Section IV.E.

B. Relevance Weighting

The prototype \( P \) is further leveraged to reweight the multi-scale query feature map \( H_{Q0} \), so that the irrelevant pixels’ features are pre-suppressed before the multi-scale fusion and distance measure. The weights are calculated based on the relevance \( R \) between \( H_{Q0} \) and \( P \),

\[
R_{ij} = 1 + \frac{w_1 H_{Q(i,j)} \cdot w_1 P}{\|w_1 H_{Q(i,j)}\|_2 \times \|w_1 P\|_2}
\]

where \( w_1 \) and \( w_2 \) are two learnable weights, used to compressed \( H_{Q0} \) and \( P \) to 64 channels. The cosine similarity between the two compressed vectors is calculated, so that \( R_{ij} \) ranges from 0 to 2 as the relevance increases. Afterwards, the relevance map \( R \) is used to reweight \( H_{Q0} \) by element-wise product \( R \times H_{Q0} \). Thus, when a pixel \( H_{Q(i,j)} \) is irrelevant to the object type, \( R_{ij} \) approaches zero and this pixel’s feature is suppressed.

---

*Fig. 2. CPieNet architecture. The CPI is labeled by red dashed line.*
C. Training Loss

At each training step, a support-query image pair \( \{I_s, I_q\} \) of the same object as well as the support CPI map \( C_s \) are fed into the CPIeNet, to predict the query CPI map \( C_q \). Because CPI is narrow, standard cross-entropy (CE) loss cannot handle the imbalance in pixel numbers of the foreground and background. The weighted CE loss has a hyper-parameter weight to tune. We utilize the Dice loss to supervise the learning of CPI extraction, which can lead to sharp contour prediction and has no extra hyper-parameter [26].

\[
L = 1 - \sum_{i,j} 2C_{s(i,j)}C_{q(i,j)}^* \sum_{i,j} C_{s(i,j)}^* + \sum_{i,j} C_{q(i,j)}^* + \tau
\]  

where \( C_q \) is the ground-truth of query CPI map. The small positive constant \( \tau \) is used to stabilize the computation.

D. Support-query Sample Pair Generation

Since support-query sample pairs are required to train CPIeNet, instead of collecting an image pair for every object and labelling them coordinately, we utilize the online public images to build datasset, which covers a wider range of object types with a much lower cost. Each image is annotated individually, then used to generate a sample pair automatically.

An \( H \times W \) image and one of its CPI annotations are called a raw sample \( \{I_R, C_R\} \). To generate a support-query sample pair from a raw sample, we customized random data augmentation to mimic imaging condition variation, as implemented by the following steps. For the convenience, the default parameters are directly presented here, which can be adjusted in practice.

1) Mix-up: \( I_R \) is mixed with another randomly-selected image \( I_1 \) using weighted sum \( I_{mix} = (1-\gamma_{mix})I_R + \gamma_{mix}I_1 \), where \( \gamma_{mix} \in [0,0.3] \) is a random weight. Thus random shade is overlapped on the object.

2) Cutout & Patch: Select another image \( I_2 \) and cut out a patch \( I_{pad} \) from its \( H/2 \times W/2 \) center area randomly. The width and height of \( I_{pad} \) randomly range within \( [W/5, W/2] \) and \( [H/5, H/2] \), respectively. Then \( I_{pad} \) is shrunk to be smaller than \( H/3 \times W/3 \), and put in \( I_{mix} \) at a random position without covering the CPI, to mimic a stuff near the object.

3) Padding: Select another image \( I_3 \), center-crop a \( H/2 \times W/2 \) patch from it, and resize the patch to 1.4\( H \times 1.4W \), which is labeled as \( I_{pad} \). Then \( I_{mix} \) is overlaid at the center of \( I_{pad} \). Meanwhile \( C_R \) is padded to 1.4\( H \times 1.4W \) with zeros. Thus, \( I_{pad} \) has not only the object but also other stuff near the boundary.

4) Data augmentation: The ordinary data augmentation is used to mimic the translation, rotation, scaling, illumination change and color change, which is introduced in Section V.A. Thus, \( I_{pad} \) and padded \( C_R \) are transformed to \( I_{aug} \) and \( C_{aug} \), respectively, whose sizes are 1.4\( H \times 1.4W \).

5) Cropping: \( I_{aug} \) and \( C_{aug} \) are cropped randomly down to the original size of \( H \times W \). Note that the CPI in \( C_{aug} \) should not be cropped off. Since Step 3 provides stuff near the boundary, after random cropping the final \( I_{aug} \) might still have other stuff occurred near the image boundary, to mimic the background change cause by translation.

6) Repeat the Steps 1-5 for twice. In the 1\( \text{st} \) time the \( \{I_{aug}, C_{aug}\} \) is produced as the support sample \( \{C_s, I_s\} \), and in the 2\( \text{nd} \) time as the query sample \( \{C_q, I_q\} \).

E. Gabor-Filters based Non-Maximum Suppression

The ideal contour for vision measurement should be single-pixel wide. However, the raw CPI given by CPIeNet is sharp but not guaranteed single-pixel wide, because the convolution operation leads to diffused edge. In [16] and [27], non-maximum suppression (NMS) is applied along the edge’s normal direction, which is estimated by the local gradient\(^1\) to sharpen the raw contour.

Comparing to the ordinary edge and contour that have many irregular or curly parts, the CPI in our task has regular shape, either LS or CA. Therefore, we proposed Gabor-filters based NMS (GF-NMS) to improve the contour thinning performance for CPI. Gabor filter is featured by its sensitivity to direction and spatial frequency [28]. The Gabor kernel \( g \) is determined by the five parameters: standard deviation \( \sigma \), normal direction \( \theta_n \), wavelength \( \lambda_n \), aspect ratio \( \gamma_n \), and phase offset \( \psi_n \). By selecting proper parameters and the four directions \( \theta_n = [0^\circ, 45^\circ, 90^\circ, 135^\circ] \), the four truncated Gabor kernels \( \{g_1, g_2, g_3, g_4\} \) with the size 9\( \times \)9 and ridge-shape are constructed, as visualized in Fig. 3(a).

The proposed GF-NMS is described in Algorithm 1. As illustrated in Fig. 3, after the Gabor filtering running on GPU, the four-direction response maps are obtained. For each pixel, the direction with the strongest response is regarded as the approximate normal direction of the local contour. With the direction map, NMS is conducted within the 8-pixel neighborhood along the approximate normal direction.

\(^1\) https://github.com/pdollar/edges

---

**Algorithm 1: Gabor-Filters based NMS**

**Input:** Contour map \( C^{cont} \), Gabor kernels \( \{g_1, g_2, g_3, g_4\} \), threshold \( g_0 \).

**Output:** Thinned contour map \( C_T \).

1. Smooth: \( S = g_b \ast C \); \# \( g_b \) is a Gaussian kernel; 
2. Initialize \( D^{bw}_{H,W} \) and \( C^{bw}_{H,W} \) with zeros; 
3. for each \( g_k \) \((k=1,2,3,4)\), \# 4-direction Gabor filtering \( g_k \rightarrow g_k \ast S \); 
4. for each pixel \( D_{ij} \) of \( D \), \# Obtain direction map \( D_{ij} = \text{argmax} (g_1, g_2, g_3, g_4) \); \# max(1,2,3,4); 
if \( C_{ij} = 0 \), then \( D_{ij} = 0 \); 
5. for each pixel \( C_{R(i,j)} \) of \( C_r \), \# NMS 
if \( D_{ij} = 1 \) and \( S_{ij} \geq \max(S_{i-1,j}, S_{i+1,j}, S_{i,j-1}, S_{i,j+1}) \), then \( C_{R(i,j)} = C_{ij} \); 
if \( D_{ij} = 2 \) and \( S_{ij} \geq \max(S_{i-1,j}, S_{i+1,j}, S_{i,j-1}, S_{i,j+1}) \), then \( C_{R(i,j)} = C_{ij} \); 
if \( D_{ij} = 3 \) and \( S_{ij} \geq \max(S_{i-1,j}, S_{i+1,j}, S_{i,j-1}, S_{i,j+1}) \), then \( C_{R(i,j)} = C_{ij} \); 
if \( D_{ij} = 4 \) and \( S_{ij} \geq \max(S_{i-1,j}, S_{i+1,j}, S_{i,j-1}, S_{i,j+1}) \), then \( C_{R(i,j)} = C_{ij} \); 
6. return \( C_T \).
V. EXPERIMENTS

A. OCP Dataset and Sample Pair Generation

We collected 2307 online public images containing various object types, including mechanical parts, digital products, industrial devices, household items, electronic components, containers, etc. Each image was resized to 320×320, and had at least one CPI. As a result, 1807 images that had 4844 LS samples and 622 CA samples were used for training, and the remaining 500 images that had 1297 LS samples and 186 CA samples were used for testing.

The proposed support-query sample pair generation method was executed to transform the above raw samples to sample pairs. Note that the training sample pairs were generated online at each training step, using the randomness to cover more condition variances. The test sample pairs were produced and fixed, for the fair evaluation of different methods. The ground-truth CPIs in training and test sets were 3-pixel and 1-pixel wide, respectively. The thicker CPIs were used for training because manual annotation might have slight error. The data augmentation step in Section IV.C is implemented with the imgaug library. First, affine transformation was applied, including scaling within [0.8,1.2], translation within [-0.2,0.2] of image size, in-plane rotation within [-15°,15°], shear angle within [-15°,15°], width/height change within [0.8,1.2]. Then, coarse dropout with the size percentage within [0.1,0.3] was conducted, followed by the slight changes on brightness, hue, saturation and gamma contrast.

Four examples of sample pair generation are shown in Fig. 4. Besides the ordinary image augmentation, the proposed generation method provided additional variations. As shown in Fig. 4(b), the generated query image had both the overlapping shade and another bottle stuff near the object, produced by the mix-up and cutout & patch steps, respectively. As shown in Fig. 4(a), the generated query image had different background near the boundary, realized by the padding and cropping steps.

B. ROCM Dataset

To evaluate the CPI extraction performance in the real environment, we collected the images of 15 different 3D objects with an ABB IRB-1200 industrial robot and a Basler acA2440-35uc industrial camera with an 8mm lens, as shown in Fig. 5. For each object, it was put in the camera view and remains static, then the robot arm moved actively to capture a series of images, observing the object from different viewpoints. Meanwhile, sometimes the illumination was changed and other stuff were put near the object. The images were resized to 320×320 and recorded. Thus, 15 image series including 523 images in total were obtained and annotated, providing 2188 LS samples and 334 CA samples.

Two evaluation modes were used. The first mode used the 1st frame of an image series as the support image, and the other frames as the query images. The second mode used a template image as the support image, which was captured by a consumer-grade camera when putting the object on a black pad, and the query images were the same with those of the first mode. Apparently, the template based evaluation is more challenging because the different imaging device and scene.

C. Training Details and Evaluation Metrics

The input size was 320×320. The model training is based on the Adam optimizer, with the initial learning rate of 0.0001, the batch size of 4, and the training epochs of 40. The learning rate was decayed by 0.5 every 10 epochs. The ResNet-50 backbone was pre-trained on ImageNet. Before the online sample pair generation at each training step, the image and annotation were randomly flipped vertically and horizontally. In the GF-NMS algorithm, the 5×5 Gaussian kernel had the standard deviation (SD) of 1 and the threshold g0 was 2.0. The four Gabor kernels had the SD of 2, wavelength of 9, aspect ratio of 0.3 and phase offset of 0. The hardware configuration included a 3.70GHz Intel i7-8700K CPU and two NVIDIA RTX2080ti GPUs.

Following the edge detection work [18], the maximum F-Measure (MF) at optimal dataset scale (ODS) was adopted as the evaluation metric of CPI extraction performance, regarding CPI map as edge map. The misalignment tolerance threshold was set to 0.01L, and L is the diagonal length of the map.

D. Ablation Experiments

A series of ablation experiments were conducted to validate the effectiveness of the proposed methods. The results are reported in Table I. The experiment No. 1 was regarded as the comparison baseline, which used the CPieNet with cosine distance but without relevance weighting (RW). 1) In No. 2 experiment, we replaced cosine distance with Euclidean distance, leading to the worse performance. 2) In No. 3 experiment, we simplified the support-query sample pair generation during training, by using only the ordinary data.

---

2 https://github.com/aleju/imgaug.
augmentation in Step 4 of Section IV.D. As a result, the performance degraded significantly.

3) In No. 4 experiment, the predicted raw contour was directly used for evaluation without using the proposed GF-NMS, resulting in the lower MF-ODS scores, which showed the necessity of contour thinning. 4) In No. 5 experiment, adding the proposed RW module, the MF-ODS scores were further increased.

CPieNet presented the real-time inference speed with the 320x320 input size, as reported in Table I. Comparing the runtime in No. 1 and No. 2 experiments, adding GF-NMS module only increased the runtime by less than 1ms. Comparing the runtime in No. 1 and No. 5 experiments, adding RW module also just increased the runtime by less than 1ms.

In Fig. 6, five examples of CPieNet inference on OCP dataset are visualized. It is shown by the relevance maps that the proposed RW module could automatically learn to highlight the pixels more relevant to the target CPI. With the relevant pixels' features enhanced and the irrelevant pixels' features suppressed, the following distance measure can be more concentrated on the discrimination of similar but different contour parts. As shown in the 2nd row, CPieNet without RW failed to distinguish the line segments with and without screw thread, leading to the false positive extraction. As shown in the 3rd and 4th rows, CPieNet without RW also presented false positive contours near the ground-truth CPI. In comparison, CPieNet with RW extracted the clearer CPIs.

We visualize the batch of CPIs extraction results with CPieNet on ROCM dataset in Fig. 7. As shown in the first row, the support image was the 1st frame of the image sequence. The two query images contained the same silicon chip object, but were captured with changed view points and illumination. CPieNet extracted the four CPIs from the query images, which could be used to localize the silicon chip. As shown in the second row, the query images were the same with those in the first row, and the support image was changed to a template image captured with different camera and background. As a result, the CPI extraction performance degraded and some pixels of CPI were missed. The overall experiments revealed that CPieNet sometimes failed when background or view angle changed significantly, because the generated training sample pairs could not cover all the variations in real environment.

E. Comparison Experiments

To the best of our knowledge, CPI extraction task has not been tackled by previous methods. Therefore, we re-implemented the core methods in the related works [10, 20,
The re-implementation details different from the original versions are as follows.

1) **CANet**: The feature extractor to obtain $\mathcal{H}_S$ and $\mathcal{H}_Q$ was the same with CPieNet’s. $\mathcal{H}_Q$ and $\mathcal{P}$ are concatenated, then fused by a 3×3 convolutional layer with 128 channels and the dilated rate of 2. The intermediate convolutional layers in the iterative optimize module all had 128 channels, and the iterative refinement was repeated by four times. 2) **PANet**: The feature extractor to obtain $\mathcal{H}_S$ and $\mathcal{H}_Q$ was the same with CPieNet’s. 3) **PANet-sigmoid**: we also investigated the PANet variant that only used the foreground prototype and its output activation was realized by Eq. (4). 4) **Feature weighting**: The feature extractor to obtain $\mathcal{H}_S$ and $\mathcal{H}_Q$ was the same with CPieNet’s. 5) **SG-One**: The feature extractor to obtain $\mathcal{H}_S$ and $\mathcal{H}_Q$ was the same with CPieNet’s. The guidance branch had three 3×3 convolutional layers with 128 channels and the stride of 1×1. The segmentation branch was composed by 1×1 convolutional layers with 128 channels.

The same training configuration and GF-NMS was applied to all the methods. The evaluation results on the two dataset are reported in Table II. CANet had the best CPI extraction performance on the OCP test dataset. However, its performances degraded significantly on ROCM dataset, which showed that the generalization ability from OCP dataset to ROCM dataset was not satisfactory. PANet originally has two prototypes for both foreground and background, and compares the query feature’s distances to these two prototypes. However, the background prototype might have poor generalization ability, because the background usually changes. PANet-sigmoid is similar to CPieNet, only considering the

![Fig. 7. Batch of CPIs extraction on ROCM dataset. Each row show an example. Different colors mark the multiple different CPIs on the same object.](image-url)
foreground prototype, which had the significantly improved performance than the original version. In comparison, CPieNet demonstrated the best overall performance on the two datasets. Besides, we investigated the contour thinning performances of the gradient-based NMS [16,27] and the proposed GF-NMS. The results showed that GF-NMS had the better performance on OCP dataset and the ROCM dataset under the 1st frame based evaluation mode. GF-NMS cost less runtime because the main computation was on GPU. GF-NMS performed slightly worse than gradient-based NMS on ROCM dataset under the template-based evaluation mode, because some weak true positive responses were suppressed to zero in GF-NMS.

VI. CONCLUSION

Object-agnostic geometric image feature extraction is an essential step to realize object-agnostic vision measurement. Towards this target we propose the CPieNet model under the one-shot learning framework. Given an image of a novel-type object to perceive, CPieNet extracts the designated CPI from it, according to the prototype vector obtained from an annotated example image. The relevance weighting module is embedded to improve the discrimination ability by enhancing relevant pixels before dense similarity comparison. GF-NMS is proposed to thin the regular-shape CPI to one-pixel wide, considering the requirement of precise measurement. The paired training samples are generated from online public images, with lower cost and wider range of object types. The two novel datasets OCP and ROCM are built for training and evaluating the proposed model. The future work will continue to improve the robustness and accuracy of CPI extraction.

REFERENCES

[1] M. Pfanne, M. Chalon, F. Stulp, et al., “Fusing joint measurements and visual features for in-hand object pose estimation,” IEEE Robot. Autom. Lett., vol. 3, no. 4, pp. 3497-3504, 2018.
[2] X. Yin, M. Yin, L. Xie, et al., “Automatic measurement system of section geometry parameters for composite filter rods based on machine vision,” IEEE Trans. Instrum. Meas., vol. 69, no. 7, pp. 5037-5050, 2020.
[3] J. Yang, C. Wang, B. Jiang, et al., “Visual perception enabled industry intelligence: state of the art, challenges and prospects,” IEEE Trans. Ind. Informat., to be published, 2020.
[4] L. Lecrosnier, R. Bouteau, P. Vasseur, et al., “Camera pose estimation based on Phl. with a known vertical direction,” IEEE Robot. Autom. Lett., vol. 4, no. 4, pp. 3852-3859, 2019.
[5] R. G. Lins, S. N. Givigi, P. R. G. Kurka, “Vision-based measurement for localization of objects in 3-D for robotic applications,” IEEE Trans. Instrum. Meas., vol. 64, no. 11, pp. 2950-2958, 2015.
[6] S. Sun, Y. Yin, X. Wang, et al., “Robust landmark detection and position measurement based on monocular vision for autonomous aerial refueling of UAVs,” IEEE Trans. Cybern., vol. 49, no. 12, pp. 4167-4179, 2018.
[7] J. Peng, W. Xu, L. Yan, et al., “A pose measurement method of a space noncooperative target based on maximum outer contour recognition,” IEEE Trans. Aeroe. Electron. Syst., vol. 56, no. 1, pp. 512-526, 2019.
[8] S. Liu, D. Xu, F. Liu, et al., “Relative pose estimation for alignment of long cylindrical components based on microscopic vision,” IEEE/ASME Trans. Mechatronics, 2015, 21(3): 1388-1398.
[9] K. Chaudhary, K. Wada, X. Chen, et al., “Learning to segment generic handheld objects using class-agnostic deep comparison and segmentation network,” IEEE Robot. Autom. Lett., vol. 3, no. 4, pp. 3844-3851, 2018.
[10] C. Zhang, G. Lin, F. Liu, et al., “CANet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp. 5217-5226, 2019.
[11] E. Lu, W. Xie, A. Zisserman, “Class-agnostic counting,” in Proc. Asian Conf. Comput. Vis., pp. 669-684, 2018.
[12] Z. He, Z. Jiang, X. Zhao, et al., “Sparse template-based 6-D pose estimation of metal parts using a monocular camera,” IEEE Trans. Ind. Electron., vol. 67, no. 1, pp. 390-401, 2019.
[13] X. Zhang, Z. Jiang, H. Zhang, et al., “Vision-based pose estimation for textureless space objects by contour points matching,” IEEE Trans. Aerosp. Electron. Syst., vol. 54, no. 5, pp. 2342-2355, 2018.
[14] Y. Li, F. Shen, D. Zhang, et al., “Contour primitives of interest extraction method for microscopic images and its application on pose measurement,” IEEE Trans. Syst. Man Cybern. Syst., vol. 48, no. 8, pp. 1348-1359, 2018.
[15] J. Yang, B. Price, S. Cohen, et al., “Object contour detection with a fully convolutional encoder-decoder network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp. 193-202, 2016.
[16] Y. Liu, M.-M. Cheng, X. Hu, et al., “Richer convolutional features for edge detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 8, pp. 1393-1406, 2019.
[17] H. Le, L. Mai, B. Price, et al., “Interactive boundary prediction for object selection,” in Proc. Europ. Conf. Comput. Vis., pp. 18-33, 2018.
[18] Z. Yu, C. Feng, M.-Y. Liu, et al., “CASENet: Deep category-aware semantic edge detection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp. 5964-5973, 2017.
[19] S. Huang, F. Qin, P. Xiao, et al., “TP-LSD: Tri-points based line segment detector,” Proc. Europ. Conf. Comput. Vis., accepted, 2020.
[20] K. Wang, J. H. Liew, Y. Zou, et al., “PANet: Few-shot image semantic segmentation with prototype alignment,” in Proc. IEEE Int. Conf. Comput. Vis., pp. 9197-9206, 2019.
[21] K. Nguyen, S. Todorovic, “Feature weighting and boosting for few-shot segmentation,” in Proc. IEEE Int. Conf. Comput. Vis., pp. 622-631, 2019.
[22] X. Zhang, Y. Wei, Y. Yang, et al., “S2-Net: Similarity guidance network for one-shot semantic segmentation,” IEEE Trans. Cybern., vol. 50, no. 9, pp. 3855-3865, 2020.
[23] T. Hu, P. Yang, C. Zhang, et al., “Attention-based multi-context guiding for few-shot semantic segmentation,” in Proc. AAAI Conf. Artificial Intell., vol. 33, pp. 8441-8448, 2019.
[24] K. Hu, X. Zhang, S. Ren, et al., “Identity mappings in deep residual networks,” in Proc. Europ. Conf. Comput. Vis., pp. 630-645, 2016.
[25] L. C. Chen, Y. Zhu, G. Papandreou, et al., “Encoder-decoder with atrous separable convolution for semantic image segmentation,” in Proc. Europ. Conf. Comput. Vis., pp. 801-818, 2018.
[26] F. Qin, S. Lin, Y. Li, et al., “Towards better surgical instrument segmentation in endoscopic vision: multi-angle feature aggregation and contour supervision,” IEEE Robot. Autom. Lett., vol. 5, no. 4, pp. 6539-6646, 2020.
[27] P. Dollár, C. L. Zitnick, “Fast edge detection using structured forests,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 8, pp. 1558-1570, 2014.
[28] R. Mehrkota, K. R. Namaduni, N. Ranganathan, “Gabor filter-based edge detection,” Pattern Recognit., vol. 25, no. 12, pp. 1479-1494, 1992.