PPR-Net: Patch-based multi-scale pyramid registration network for defect detection of printed label

Dongming Li[0000−0001−6538−0655], Yingjian Li[0000−0002−0653−4535], Jinxing Li[0000−0001−5156−0305], and Guangming Lu*[0000−0003−1578−2634]

Harbin Institute of Technology Shenzhen, Shenzhen, China
{dongmingli_2018,hit_lyj}@126.com, {lijinxing158,luguangm}@hit.edu.cn

Abstract. Detecting defects in printed labels is essential to ensure product quality. Reference-based comparison is a potential method to challenge this task, which is widely used for defect detection. However, this method gets poor performance under large deformation, due to the lack of ability of registering the testing image with the reference image. Therefore, accurate image registration is an urgent case for defect detection of printed labels. In this paper, a patch-based multi-scale pyramid registration network (PPR-Net) is proposed. First, an image patch splitting and stitching strategy is proposed, which is scalable in image resolution. Second, a multi-scale pyramid registration module is designed to fuse multiple convolutional features to enhance the registration capability for large deformation, which gradually refines multi-scale deformation fields in a coarse-to-fine manner. Third, a distortion loss function is introduced to improve text distortions of registered images. Finally, a synthetic database is generated based on real printed labels, to simulate defective printed labels with large deformation for performance comparison. Extensive experimental results show that our method dramatically outperforms other comparable approaches.

Keywords: Printed labels · multi-features fusion · artifact · deformable registration · defect detection.

1 Introduction

Nowadays, manufacturing enterprises of electronic products are facing fierce competition, and improving product quality is crucial to enhance competitiveness. As an essential part of electronic products, printed labels are widely used. Therefore, it is of great significance to perform defect detection of printed labels during manufacturing. Up to now, one of the most widely used techniques is the reference-based comparison method. This method stores a reference image in advance and then compares it with the testing image at pixel-by-pixel or feature-by-feature level. If any pixels or features do not match, defects may exist [14, 20].

* Corresponding author.

However, most printed labels are manufactured from non-rigid materials. Due to the mechanical vibration and distortion of printed labels, large deformations are inevitably introduced. At present, there is no uniform specification to define large deformation. Here, we use the average mean square error (MSE) to evaluate large deformation. Subsequently, these deformations may result in obvious artifacts after image subtraction, increasing the difficulty of defect detection.

Based on this aforementioned analysis, accurate image registration becomes an urgent case for defect detection of printed labels. At present, image registration algorithms can be categorized into two groups: traditional and deep learning methods. Traditional methods can be roughly divided into two categories: pixel grayscale-based and feature-based registration methods. Pixel grayscale-based method is stable and usually measures the similarity of two images based on full grayscale information, such as the normalized cross-correlation registration algorithm [19]. However, the pixel grayscale-based method is often computationally expensive and sensitive to illumination. The feature-based method usually extracts image features, such as corner, edge, shape, and texture. The existing feature extraction operators mainly include SIFT operator [11], SURF operator [3], AKAZE operator [1], and so on. These feature-based methods have a low computational cost and strong robustness. However, such methods seem to be ineffective [7] for images with inconspicuous features.

In recent decades, many deep learning-based methods have been innovated to achieve image registration. There are several recent works [9, 15, 17, 18] to learn a function for image registration based on neural networks. However, most of them rely on ground truth deformation fields for training models. By contrast, our PPR-Net is an unsupervised method. Since the success of the spatial transformer network [6], unsupervised deep learning techniques have achieved state-of-the-art performance in many registration tasks. For example, Balakrishnan et al. [2] proposed an unsupervised registration network (VoxelMorph), which makes a straightforward prediction of the deformation field based on the U-net [16] structure. However, VoxelMorph is usually limited to small deformations. To solve this issue, recent works, such as DDN [13], PRDFE [21], and Dual-PRNet [5], introduced new improvements to handle large deformations. Nevertheless, few of them pay attention to the text distortion of registered images, which has a great impact on the accuracy of defect detection.

To address the above challenges, we focus on the enhancement of large deformable image registration. Our main contributions are:

1) We introduce a patch-based image splitting and stitching strategy, which is scalable in image resolution and overcomes the limited memory of GPUs.
2) We propose a multi-scale pyramid registration module with a multi-feature fusion strategy. Multi-scale deformation fields are refined gradually in a coarse-to-fine manner, to boost the registration performance for large deformations.
3) We design a novel distortion loss function, which is incorporated with shift-invariant loss, to improve text distortion of registered images.

4) We generate a synthetic database based on real printed labels, to simulate defective printed labels with large deformation. Furthermore, we evaluate the accuracy of registration from two aspects: pixel level and defect detection. Experimental results show that our method performs better than other compared methods.

The remainder of this paper is organized as follows. Section 2 presents the details of our proposed method. Section 3 presents our experimental results, followed by a conclusion in Section 4.

2 Method

2.1 The proposed framework

Fig. 1 shows the overall framework of our method. It consists of an image registration network with a patch-based image splitting and stitching strategy, followed by reference-based comparison defect detection (RCDD) to evaluate the accuracy of image registration. Fixed image, moving image and warped image represent reference image (defect-free), testing image and registered image, respectively. Patches are first extracted from fixed and moving images based on the image splitting strategy, then fed into the registration network. After image registration and stitching, a final registered image is obtained. In this paper, we focus on image registration, which is used for defect detection of printed labels.

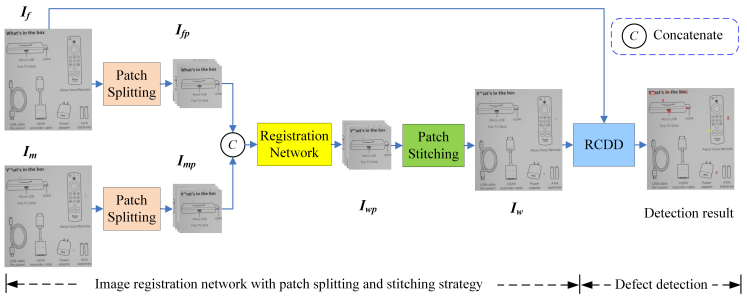
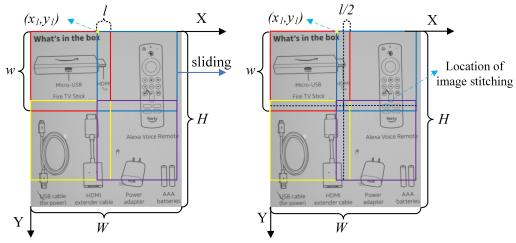


Fig. 1: The proposed framework for image registration. I_f and I_m are the fixed and moving image, which are split into patches I_{fp} and I_{mp}. I_{wp} are patches warped by registration. I_w is the image stitched from I_{wp}.

2.2 Patch splitting and stitching strategy

Table 1 shows that resolutions of testing images are different, and a common approach is to resize them to the same size before image registration. However, resizing the image is not recommended, as it may result in some inconspicuous

defects being missed. In addition, GPU memory is limited in the real-world industrial scene. To deal with these problems, we introduce a patch-based image splitting and stitching strategy, as shown in Fig. 2. For image splitting, slide patches are generated by sliding a window on fixed and moving images, respectively. A sliding window starts from the red frame and slides $w-l$ pixels per step along the X axis. After reaching the right edge of the testing image, the sliding window returns to the location of the yellow frame and slides along the X axis again. q is the number of slide patches, and $q = \lceil (W-l)/(w-l) \rceil * \lceil (H-l)/(w-l) \rceil$. $W \times H$ and $w \times w$ are the size of the testing image and sliding window, respectively. l is the overlap pixels. The location of image stitching is at half of the overlap of two adjacent slide patches. Algorithm 1 and 2 gives the details.



(a) Image splitting (b) Image stitching

Fig. 2: Patch splitting and stitching strategy. (x_1, y_1) is the upper-left coordinate of the sliding window.

Algorithm 1: Image splitting

Input: I_m, I_f, l, w, W, H, q
Output: Sets $\mathbb{P}^m$, $\mathbb{P}^f$, $\mathbb{C}$, contain moving, fixed patches and corresponding coordinate.

for $i \in [0, \lceil (H-l)/(w-l) \rceil - 1]$ **do**
  if $((w-l) * i + w) > H$ **then**
    $y_1 = H - w$
  else
    $y_1 = (w-l) * i$
  end
  for $j \in [0, \lceil (W-l)/(w-l) \rceil - 1]$ **do**
    if $((w-l) * j + w) > W$ **then**
      $x_1 = W - w$
    else
      $x_1 = (w-l) * j$
    end
    Put a $w \times w$ sliding window at (x_1, y_1) of I_m and I_f, put the areas of the two sliding windows into $\mathbb{P}^m$ and $\mathbb{P}^f$. Put (y_1, x_1) into $\mathbb{C}$.
  end
end
Output $\mathbb{P}^m = \{P_1^m, P_2^m, ..., P_q^m\}$, $\mathbb{P}^f = \{P_1^f, P_2^f, ..., P_q^f\}$, $\mathbb{C} = \{C_1, C_2, ..., C_q\}$.

Algorithm 2: Image stitching

Input: I_{wp}^k, $\mathbb{C}$, l, w, q. I_{wp}^k means the k-th warped image patch, $k = 1, ..., q$.
Output: I_w

```
for C_k ∈ ℂ do
    y = C_k[0], x = C_k[1] // C_k[0],C_k[1] meaning the coordinates of Y and X axes.
    if (y == 0 and x == 0) then
        I_w[0 : w − l/2, 0 : w − l/2] = I_wp^k[0 : w − l/2, 0 : w − l/2];
    else if (y == 0 and x ≠ 0) then
        I_w[y : y + w − l/2, x + l/2 : x + w] = I_wp^k[0 : w − l/2, l/2 : w];
    else if (y ≠ 0 and x == 0) then
        I_w[y + l/2 : y + w, x : x + w − l/2] = I_wp^k[l/2 : w, 0 : w − l/2];
    else
        I_w[y + l/2 : y + w, x + l/2 : x + w] = I_wp^k[l/2 : w, l/2 : w];
    end
end
Output the final warped image I_w.
```

2.3 Multi-scale pyramid registration network

Given a pair of fixed image and moving image which are defined over a $2D$ spatial domain $\Omega \subset \mathcal{R}^2$, image registration seeks to find a coordinate transform between the image pair. A convolutional neural network (CNN) is adopt to model a function f to estimate the optimal deformation field $\Phi = f_\theta(I_{fp}, I_{mp})$. θ represents the learnable parameters of function f. I_{fp} and I_{mp} are corresponding patches extracted from the fixed and moving image. Fig. 3 illustrates the architecture of our proposed registration network, which consists of an encoder module, a decoder module, and a multi-scale pyramid module.

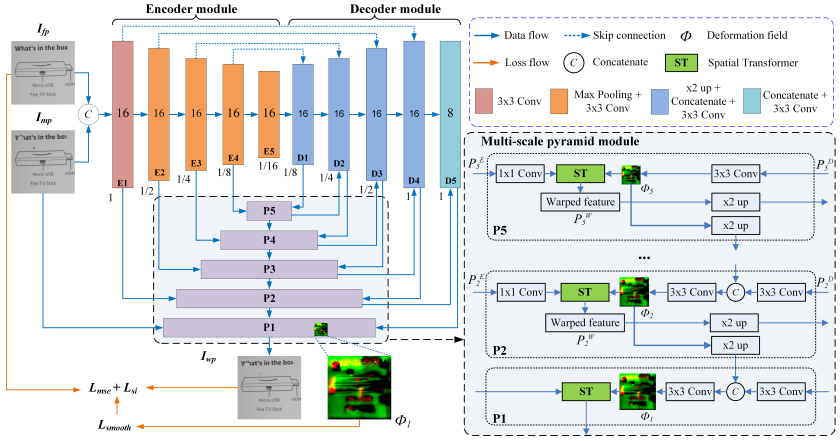


Fig. 3: The architecture of our proposed registration network. P_i means the i-th pyramid layer. Details of the multi-scale pyramid module are shown on the right.

Encoder module. We adopt an encoder with the same architecture as VoxelMorph [2], which consists of five encoding blocks, named E_1, ..., E_5. Fixed and moving patches are concatenated and fed into the first encoding block, with a 3×3 convolution. The rest encoding blocks consist of a 3×3 down-sampling convolution with a stride of 2, which reduces the resolution of input image pairs by a factor of 16 in total.

Decoder module. There are also five decoding blocks in the decoder module, named D_1, ..., D_5. We apply skip connections to the corresponding encoding and decoding blocks. The first four decoding blocks consist of a 3×3 up-sampling convolution with a stride of 2, followed by a concatenation and 3×3 convolution. Unlike VoxelMorph [2], to allow the network to learn stronger and more discriminative convolutional features, we fuse the multi-scale warped features into the corresponding decoding blocks. The warped features are generated by spatial transformation on the i-th pyramid layer, as shown on the right of Fig. 3.

Multi-scale pyramid module. VoxelMorph [2] only generates a single deformation field, which limits its ability to deal with large deformations. To overcome this, we introduce a multi-scale pyramid module with multi-features fusion, as follows: 1) We apply 1×1 convolution to the corresponding encoding blocks to extract multi-scale convolutional features, as the first input to spatial transformer (ST). 2) We estimate multi-resolution deformation fields Φ_i on the i-th pyramid layer, as the second input to ST. Especially, we adopt a multi-resolution deformation field fusion (MDFF) mechanism: a low-resolution deformation field is gradually fused into the high-resolution one by a series of operations, such as up-sampling, concatenation, and convolution. 3) Multi-scale warped features fusion (MWFF) mechanism. We perform spatial transform by ST on each pyramid layer and obtain corresponding warped features, which are concatenated into corresponding decoding blocks for further convolutional feature fusion.

Specifically, the multi-resolution deformation fields are defined as:

$$\Phi_i = \begin{cases} Conv^{3\times3}\left(Cat\left(Up(\Phi_{i+1}), Conv^{3\times3}\left(P_i^D\right)\right)\right), & i = 1,2,3,4 \\ Conv^{3\times3}\left(P_i^D\right), & i = 5 \end{cases} \tag{1}$$

where Φ_i indicates the i-th deformation field, $Conv^{3\times3}(.)$ denotes a 3×3 convolution, $Cat(.,.)$ is the concatenation operation, $Up(.)$ represents a up-sampling with a factor of 2, P_i^D represents the i-th decoding feature generated by the decoder module and fed into the i-th pyramid layer. Besides, the multi-scale warped features are defined as follows:

$$P_i^W = ST\left(Conv^{1\times1}\left(P_i^E\right), \Phi_i\right), i = 2, ..., 5 \tag{2}$$

where P_i^W represents the i-th warped feature on the i-th pyramid layer, $ST(.,.)$ is a spatial transformer function implemented by VoxelMorph [2], $Conv^{1\times1}(.)$ indicates a 1×1 convolution, P_i^E represents the i-th encoding feature generated by the encoder module.

The above multi-scale features P_i^E, P_i^D and P_i^W are implemented sequentially to estimate the final deformation field, which fuses multiple convolutional features with multi-scale deformation fields gradually in a coarse-to-fine manner. Finally, the warped image patch is obtained by:

$$I_{wp} = ST\left(I_{mp}, \Phi_1\right) \tag{3}$$

where Φ_1 is the final deformation field.

2.4 Loss function

Based on our observations, text distortions may occur after image registration, which directly affect the accuracy of subsequent defect detection. To this end, we design a distortion loss L_{dist}, which consists of the MSE loss L_{mse}, the local spatial variation loss L_{smooth}, and the shift-invariant loss L_{si}. The distortion loss is calculated as follows:

$$L_{dist}(I_{fp}, I_{wp}, \Phi_1) = L_{mse}(I_{fp}, I_{wp}) + \lambda_1(L_{smooth}(\Phi_1) + L_{si}(I_{fp}, I_{wp})) \tag{4}$$

whered λ_1 is a regularization trade-off parameter. The L_{mse} and L_{smooth} are defined as follows:

$$L_{mse}(I_{fp}, I_{wp}) = \frac{1}{n}\sum_{i}^{n}(I_{fp}(i) - I_{wp}(i))^2 \tag{5}$$

$$L_{smooth}(\Phi_1) = \sum \| \bigtriangledown (\Phi_1)\|^2 \tag{6}$$

where $L_{mse}(.,.)$ measures image similarity between its two inputs, and n is the number of elements in I_{wp}. I_{fp} is the corresponding patch extracted from the fixed image, and I_{wp} is the image patch warped by registration. $L_{smooth}(.)$ is a regularization constraint on the final deformation field Φ_1 to enforce spatial smoothness, $\bigtriangledown$ represents the spatial gradients. Motivated by [12], we also use a shift-invariant loss L_{si} to handle the shift problem, which is defined as follows:

$$L_{si}(I_{fp}, I_{wp}) = \frac{1}{n}\sum_{i}|d_i| - \frac{\lambda_2}{n}|\sum_{i} d_i| \tag{7}$$

where $d_i = I_{fp}(i) - I_{wp}(i)$, $I_{fp}(i)$ is the corresponding ground truth value and $I_{wp}(i)$ is the predicted value at index i, and λ_2 controls the strength of the second term. Notably, the L_{si} does not care about the absolute value of $I_{wp}(i)$. It enforces that the difference between $I_{wp}(i)$ and $I_{wp}(j)$ should be close to that between $I_{fp}(i)$ and $I_{fp}(j)$. Therefore, L_{si} is helpful to alleviate the influence of shift.

2.5 Reference-based comparison defect detection (RCDD)

As illustrated in Section 1, RCDD is one of the widely used method to detect defects of printed labels. In this paper, we first utilize PPR-Net to perform image registration, then conduct RCDD. Fig. 4 shows the process of RCDD. The input is a pair of fixed and warped images. First, a median filter is applied for denoising. Subsequently, the warped image is subtracted from the fixed image to get a difference image. Second, a morphological opening operation with a 3×3 structure element is performed on the difference image for further denoising. Finally, a low-threshold filter with threshold T_{filter} is applied to binarize the image, where a pixel value larger than T_{filter} is set to 255, otherwise 0. In the defect discrimination stage, if the area of a white pixel region in the binary image is larger than the threshold T_{area}, this region will be judged as a defect. Based on our observation, a pixel value lower than 20 is difficult to be distinguished by human beings, and defects with a pixel area less than 5 can be ignored. Therefore, we set T_{filter} and T_{area} to 20 and 5, respectively.

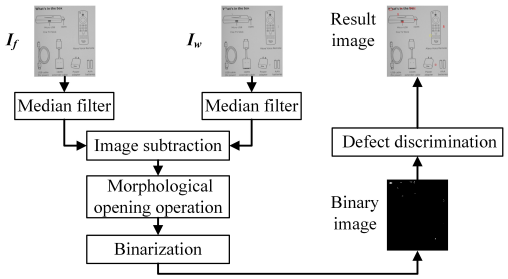


Fig. 4: The process of reference-based comparison defect detection.

2.6 Discussion

Herein, we discuss the differences between our PPR-Net and other compared deep learning methods, such as VoxelMorph [2], DDN [13], and PRDFE [21].

1) Difference with VoxelMorph [2] and DDN [13]: Both VoxelMorph and DDN predict only a single deformation field, which limits their ability to deal with large deformations. Differently, PPR-Net can predict multi-scale deformation fields based on the multi-level pyramid, which refines the deformation fields gradually and boosts the registration performance of large deformations. In addition, VoxelMorph takes the entire image as input rather than patches, which can not handle the images with different sizes. By contrast, PPR-Net uses a patch-based image splitting and stitching strategy, with better scalability of image resolution.

2) Difference with PRDFE [21]: The methods of generating deformation fields are different. PRDFE uses the warped moving and fixed features to estimate their residual deformation field at each pyramid scale. Differently, PPR-Net adopts the multi-scale warped features fusion (MWFF) mechanism to fuse warped features into corresponding decoding blocks, which further enhances the registration robustness of large deformation. Furthermore, PPR-Net is more accurate than PRDFE with less registration time, which will be reported in the experimental section.
3) All above compared methods ignore the text distortion in registered images, which may result in a high false positive rate. Differently, we introduce a novel distortion loss function incorporated with a shift-invariant loss to penalize the text distortion, which helps to reduce the false positive rate.

3 Experiments

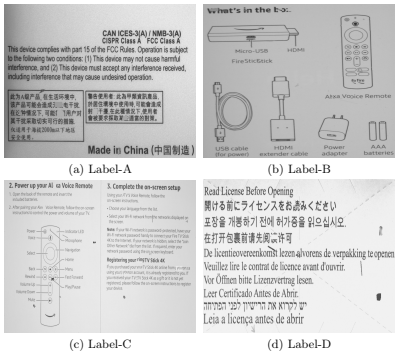
3.1 Dataset generation

As illustrated in Section 1, printed label images captured from the production line may exhibit large deformation compared with the reference image. Thus, to evaluate the performance of image registration, we need an industrial printed labels database with large deformations and defects. However, there is no such public database. To this end, we create a synthetic database in which deformable images are generated via the perturbed mesh generation algorithm [12]. We adjust the parameters d and α in [12] empirically to get proper images. Table 1 shows the summary of 10 training sets in our database. Each training set consists of one reference image and 200 synthetic images with large deformations. During the training stage, we randomly extract image patches with a size of 512×512 from synthetic images and corresponding reference images.

For testing sets, we collect four kinds of defect-free printed labels from a factory. To evaluate the registration performance with defect samples, we randomly generate 10 artificial defects for each image based on [10]. Table 1 gives a summary of our database and Fig. 5 shows some representative samples in testing sets.

Table 1: Summary of our database.

Dataset	Dataset Name	Image size $(W \times H \times C)$	Number of images
Training set	Consr	$1210 \times 1396 \times 1$	201
	Devr	$2482 \times 1457 \times 1$	201
	Donr	$3300 \times 1316 \times 1$	201
	Jackr	$2694 \times 1568 \times 1$	201
	Nomr	$1644 \times 1414 \times 1$	201
	Tvboxr	$1267 \times 1214 \times 1$	201
	Ubiqr	$900 \times 873 \times 1$	201
	Warr	$833 \times 833 \times 1$	201
	Xg2r	$1338 \times 1338 \times 1$	201
Testing set	Label-A	$1092 \times 714 \times 1$	1,475
	Label-B	$969 \times 1133 \times 1$	1,325
	Label-C	$2774 \times 1509 \times 1$	1,400
	Label-D	$1389 \times 680 \times 1$	1,472



(a) Label-A (b) Label-B (c) Label-C (d) Label-D

Fig. 5: Some representative samples in testing sets.

3.2 Evaluation metrics

In this paper, we evaluate the registration performance based on three metrics: MSE, F_1 score, and registration time. Among them, MSE is introduced to evaluate the registration accuracy at the pixel level, and the F_1 score is used for defect detection. For RCDD, better registration means better defect detection. The MSE and F_1 score are defined as follows:

$$MSE(I_f, I_w) = \frac{1}{N}\sum_{i}^{N}\left(I_f\left(i\right) - I_w\left(i\right)\right)^2 \tag{8}$$

$$F_1 = 2PR/\left(R + P\right) \tag{9}$$

where N is the number of elements in I_w, $R = TP/\left(TP + FN\right)$, R means the recall, $P = TP/\left(TP + FP\right)$, P means the precision, true positive (TP) indicates real defects that are correctly predicted as defects, false positive (FP) represents the non-defects that are incorrectly predicted as defects, and false negative (FN) means real defects that are incorrectly identified as non-defects. A defect is determined if the intersection over union (IoU) with the ground truth box is larger than 0.001.

3.3 Implementation details

The network is implemented using Keras [4], and trained with the Adam optimizer [8] with a learning rate of 0.001 on an Nvidia RTX2080 Ti GPU. The min-batch size is set to 4. During the training and testing stages, the image patch size $w \times w$ is set to 512×512. The overlap size l is set to 16. The generation of the synthetic dataset and defect detection are conducted in Python 3.6 with OpenCV. The channel numbers of encoding and decoding blocks are $[16, 16, 16, 16, 16]$ and $[16, 16, 16, 16, 8]$.

3.4 The parameter choice of $\lambda 1$ and $\lambda 2$

To verify how the values of $\lambda 1$ and $\lambda 2$ affect registration performance, we conducted experiments by varying these two parameters. As shown in Table 2, we evaluate the set $(\lambda 1, \lambda 2)$ in {(0.05, 0.05), (0.05, 0.10), (0.05, 0.50), (0.05, 1.00), (0.10, 0.10), (0.10, 0.50), (0.10, 1.00), (0.50, 0.10)} on all testing sets, where the average MSE varies from 7.059 to 11.471. The best result is achieved by $(0.05, 0.10)$. Therefore, we choose the best-optimized set $(\lambda 1 = 0.05, \lambda 2 = 0.10)$.

3.5 Comparison with baselines and time complexity

MSE is utilized to evaluate image registration performance at the pixel level. Table 3 demonstrates that our method has the lowest average MSE (7.059) and gets the lowest value for each testing set. PRDFE (8.492) and VoxelMorph (9.377) are close to our method. By contrast, the average MSE of traditional

Table 2: Effect of varying parameters $\lambda 1$ and $\lambda 2$ on MSE ($\times 10^{-4}$) (Bold: best).

$\lambda 1$	$\lambda 2$	Label-A	Label-B	Label-C	Label-D	Average MSE
0.05	0.05	17.170	5.460	3.961	11.928	9.630
0.05	**0.10**	**13.812**	**3.252**	**2.625**	**8.548**	**7.059**
0.05	0.50	15.245	4.339	3.095	10.287	8.242
0.05	1.00	15.212	3.816	2.833	10.311	8.043
0.10	0.10	16.494	5.393	3.910	11.713	9.378
0.10	0.50	15.655	4.424	3.203	10.880	8.541
0.10	1.00	19.367	6.698	5.011	14.806	11.471
0.50	0.10	17.804	6.793	4.823	13.691	10.778

methods, such as SIFT (23.728), SURF (23.134), and AKAZE (30.806), are relatively high.

In terms of F_1 score, Table 4 shows several existing methods to illustrate the effectiveness of our proposed method. Our method yields the highest average F_1 score of 0.7916, which is a new state-of-the-art performance. Further, our method clearly outperforms these existing methods, for PRDFE (0.6520), VoxelMorph (0.6378), SIFT (0.4227), SURF (0.4210), AKAZE (0.3682) and DDN (0.2629) with improvements of 13.96%, 15.38%, 36.89%, 37.06%, 42.34%, and 52.87%, respectively. In addition, our method also achieves the highest F_1 score for each testing set.

Table 5 shows the comparison of registration time. Traditional methods take a longer time than deep learning methods. Although VoxelMorph achieves the best result (0.3087 s), our registration time (0.3913 s) is close to VoxelMorph which is still in real-time.

Table 3: Comparison on MSE ($\times 10^{-4}$) (Bold: best).

Method	Label-A	Label-B	Label-C	Label-D	Average MSE
SIFT [11]	34.682	16.133	14.234	29.864	23.728
SURF [3]	34.027	15.575	13.678	29.256	23.134
AKAZE [1]	38.823	17.550	34.414	32.436	30.806
VoxelMorph [2]	17.396	4.877	3.832	11.402	9.377
DDN [13]	30.030	11.100	11.139	16.546	17.204
PRDFE [21]	15.530	4.724	3.539	10.173	8.492
PPR-Net	**13.812**	**3.252**	**2.625**	**8.548**	**7.059**

Table 4: Comparison in terms of average F_1 score (Bold: best).

Method	Label-A	Label-B	Label-C	Label-D	Average F_1
SIFT [11]	0.6275	0.3830	0.1545	0.5257	0.4227
SURF [3]	0.6262	0.3747	0.1520	0.5312	0.4210
AKAZE [1]	0.5615	0.3626	0.1425	0.4061	0.3682
VoxelMorph [2]	0.6477	0.6890	0.4145	0.8000	0.6378
DDN [13]	0.2722	0.3896	0.0677	0.3221	0.2629
PRDFE [21]	0.7203	0.6867	0.4258	0.7753	0.6520
PPR-Net	**0.7462**	**0.8571**	**0.7004**	**0.8625**	**0.7916**

Table 5: Comparison in terms of registration time (s) (Bold: best).

Method	Label-A	Label-B	Label-C	Label-D	Average time (s)
SIFT [11]	0.9877	0.6125	3.0090	0.6807	1.3225
SURF [3]	0.6032	0.4320	2.4815	0.5781	1.0237
AKAZE [1]	0.4636	0.5166	1.7136	0.5161	0.8025
VoxelMorph [2]	**0.1665**	**0.1679**	**0.7037**	**0.1965**	**0.3087**
DDN [13]	0.1731	0.1752	0.7854	0.2190	0.3382
PRDFE [21]	0.2359	0.2367	0.9437	0.2168	0.4083
PPR-Net	0.1996	0.2236	0.8997	0.2421	0.3913

3.6 Ablation studies for PPR-Net

For PPR-Net, we improve the distortion loss function L_{dist} and the multi-scale pyramid module with multi-features fusion. To figure out the performance of each part, we discard some parts of PPR-Net and get some models. Table 6 shows the settings for different models. Model "PPR-Net-w/o-L_{si}" means removing the shift-invariant loss L_{si} from the distortion loss L_{dist}. Model "PPR-Net-w/o-MDFF" represents PPR-Net without the gradual fusion from the low-resolution deformation field to the high-resolution one. Model "PPR-Net-w/o-MWFF" means removing warped features at pyramid layers from P_2 to P_5. Model "PPR-Net-Single-scale" represents that the multi-scale pyramid module is changed to single-scale by removing pyramid layers from P_2 to P_5.

Table 7 and Table 8 show the comparison results in terms of MSE and F_1 score. After the pyramid module is changed from multi-scale to single-scale, the average MSE increases significantly from 7.059 to 8.973 and the F_1 score decreases from 0.7916 to 0.6440. After removing MWFF, the average MSE increases from 7.059 to 8.951, and the F_1 score drops from 0.7916 to 0.6609. Therefore, the most significant improvement is the multi-scale pyramid, and the second one is the MWFF. Meanwhile, compared with other models, our PPR-Net has the best MSE and F_1 score on all testing sets.

Table 6: The settings of different models.

Model	L_{si}	Multi-scale pyramid		Single-scale pyramid
		Multi-resolution deformation fields fusion (MDFF)	Multi-scale warped features fusion (MWFF)	
PPR-Net-w/o-L_{si}	No	Yes	Yes	No
PPR-Net-w/o-MDFF	Yes	No	Yes	No
PPR-Net-w/o-MWFF	Yes	Yes	No	No
PPR-Net-Single-scale	Yes	No	No	Yes
PPR-Net	Yes	Yes	Yes	No

3.7 Qualitative results

Fig. 6 shows that our PPR-Net achieves more accurate alignment than the compared methods. For SIFT, SURF, and AKAZE, a black region is introduced on the left of the registered images, resulting in false detection. Further, DDN and PRDFE get significant text distortion compared with PPR-Net. Although

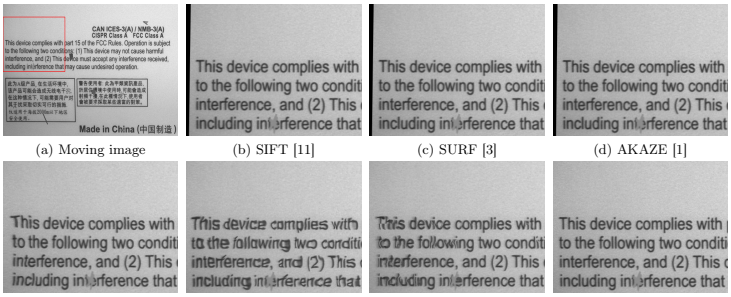
Table 7: Ablation study in terms of MSE ($\times 10^{-4}$) (Bold: best).

Model	Label-A	Label-B	Label-C	Label-D	Average MSE
PPR-Net-w/o-L_{si}	15.311	3.641	2.890	10.103	7.986
PPR-Net-w/o-MDFF	15.398	3.401	2.813	10.038	7.913
PPR-Net-w/o-MWFF	16.362	4.814	3.733	10.893	8.951
PPR-Net-Single-scale	16.344	4.704	3.887	10.955	8.973
PPR-Net	**13.812**	**3.252**	**2.625**	**8.548**	**7.059**

Table 8: Ablation study in terms of F_1 score (Bold: best).

Model	Label-A	Label-B	Label-C	Label-D	Average F_1
PPR-Net-w/o-L_{si}	0.7265	0.8567	0.6976	0.8329	0.7784
PPR-Net-w/o-MDFF	0.7432	0.8563	0.6884	0.8114	0.7748
PPR-Net-w/o-MWFF	0.7071	0.7126	0.4123	0.8117	0.6609
PPR-Net-Single-scale	0.7012	0.7208	0.3543	0.7996	0.6440
PPR-Net	**0.7462**	**0.8571**	**0.7004**	**0.8625**	**0.7916**

the performance of VoxelMorph is close to ours, a certain text distortion can be observed, such as the letter "T" of "This device" in Fig. 6e. Fig. 7 gives a comparison of subtraction images, and our method is the best with the fewest artifacts. By contrast, DDN yields significant artifacts, and the traditional methods, such as SIFT, SURF, and AKAZE, all have artifacts generated by the above black regions. Fig. 8 visualizes a comparison among compared methods and PPR-Net (Label-B), and the compared methods do exist with more FP or FN.



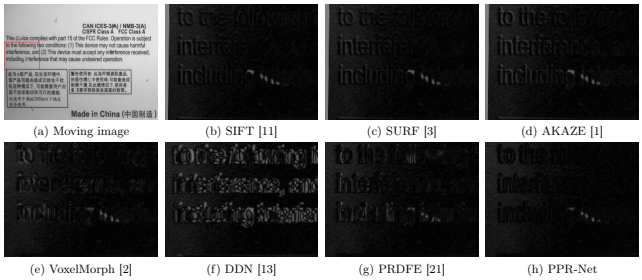
(a) Moving image (b) SIFT [11] (c) SURF [3] (d) AKAZE [1]

(e) VoxelMorph [2] (f) DDN [13] (g) PRDFE [21] (h) PPR-Net

Fig. 6: Registration performance visualization (Label-A). (a) is the moving image with an ROI indicated by the red frame, (b)-(h) are the corresponding warped images of the red frame of (a).

4 Conclusions

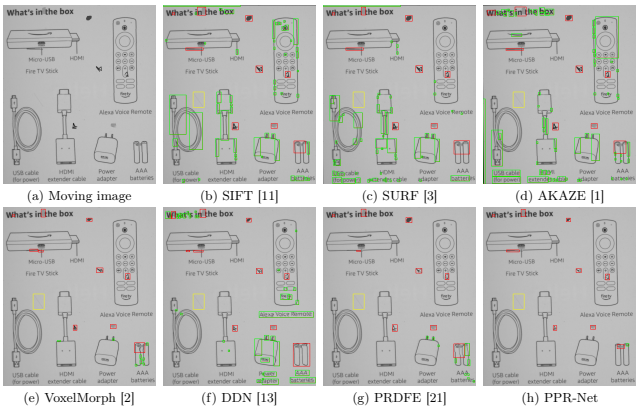
In this paper, we proposed a patch-based multi-scale pyramid registration network for printed labels with large deformation. We adopt a patch-based image splitting and stitching strategy, which can scale to larger image resolution. PPR-Net allows recurrently refining the deformation fields with multi-scale convolu-



(a) Moving image (b) SIFT [11] (c) SURF [3] (d) AKAZE [1]

(e) VoxelMorph [2] (f) DDN [13] (g) PRDFE [21] (h) PPR-Net

Fig. 7: The visualization of difference images. After image registration, warped images are subtracted from the fixed image. (a) is a moving image from Label-A. (b)-(h) are corresponding difference images of the red frame of (a) .



(a) Moving image (b) SIFT [11] (c) SURF [3] (d) AKAZE [1]

(e) VoxelMorph [2] (f) DDN [13] (g) PRDFE [21] (h) PPR-Net

Fig. 8: Visualization of defect detection results (Label-B). The results of TP, FP, and FN are marked with red, green, and yellow frames, respectively.

tional features in a coarse-to-fine manner, which helps handle large deformations. Further, we introduced a novel distortion loss function incorporated with shift-invariant loss, which can improve text distortion after image registration. Finally, we generated a synthetic database based on real printed labels and evaluated registration performance. The experimental result shows that our method outperforms other comparable approaches. The precision of reference-based comparison defect detection is significantly improved by using our registration network.

Acknowledgements This work was supported in part by NSFC fund (62176077, 62272133, 61906162), in part by the Shenzhen Colleges and Universities Stable Support Program No. GXWD20220811170100001, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2019B1515120055, in part by the Shenzhen Key Technical Project under Grant 2020N046, in part by the Shenzhen Fundamental Research Fund under Grant JCYJ20210324132210025, in part by Shenzhen Science and Technology Program (RCBS20200714114910193).

References

1. Alcantarilla, P.F., Solutions, T.: Fast explicit diffusion for accelerated features in nonlinear scale spaces. IEEE Trans. Patt. Anal. Mach. Intell **34**(7), 1281–1298 (2011)
2. Balakrishnan, G., Zhao, A., Sabuncu, M.R., Guttag, J., Dalca, A.V.: An unsupervised learning model for deformable medical image registration. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 9252–9260 (2018)
3. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. In: European conference on computer vision. pp. 404–417. Springer (2006)
4. Chollet, F., et al.: Keras. `https://github.com/fchollet/keras` (2015)
5. Hu, X., Kang, M., Huang, W., Scott, M.R., Wiest, R., Reyes, M.: Dual-stream pyramid registration network. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 382–390. Springer (2019)
6. Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. Advances in neural information processing systems **28**, 2017–2025 (2015)
7. Ke, Y., Sukthankar, R.: Pca-sift: A more distinctive representation for local image descriptors. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004. vol. 2, pp. II–II. IEEE (2004)
8. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
9. Krebs, J., Mansi, T., Delingette, H., Zhang, L., Ghesu, F.C., Miao, S., Maier, A.K., Ayache, N., Liao, R., Kamen, A.: Robust non-rigid registration through agent-based action learning. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 344–352. Springer (2017)
10. Li, D., Li, J., Fan, Y., Lu, G., Ge, J., Liu, X.: Printed label defect detection using twice gradient matching based on improved cosine similarity measure. Expert Systems with Applications **204**, 117372 (2022)
11. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International journal of computer vision **60**(2), 91–110 (2004)
12. Ma, K., Shu, Z., Bai, X., Wang, J., Samaras, D.: Docunet: Document image unwarping via a stacked u-net. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4700–4709 (2018)
13. Nazib, A., Fookes, C., Perrin, D.: Dense deformation network for high resolution tissue cleared image registration. arXiv preprint arXiv:1906.06180 (2019)
14. Peng, X., Chen, Y., Xie, J., Liu, H., Gu, C.: An intelligent online presswork defect detection method and system. In: 2010 Second International Conference on Information Technology and Computer Science. pp. 158–161. IEEE (2010)
15. Rohé, M.M., Datar, M., Heimann, T., Sermesant, M., Pennec, X.: Svf-net: learning deformable image registration using shape matching. In: International conference on medical image computing and computer-assisted intervention. pp. 266–274. Springer (2017)
16. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)
17. Sokooti, H., Vos, B.d., Berendsen, F., Lelieveldt, B.P., Išgum, I., Staring, M.: Nonrigid image registration using multi-scale 3d convolutional neural networks. In: International conference on medical image computing and computer-assisted intervention. pp. 232–239. Springer (2017)

18. Yang, X., Kwitt, R., Styner, M., Niethammer, M.: Quicksilver: Fast predictive image registration–a deep learning approach. NeuroImage **158**, 378–396 (2017)
19. Yoo, J.C., Han, T.H.: Fast normalized cross-correlation. Circuits, systems and signal processing **28**(6), 819–843 (2009)
20. Zhang, E., Chen, Y., Gao, M., Duan, J., Jing, C.: Automatic defect detection for web offset printing based on machine vision. Applied sciences **9**(17), 3598 (2019)
21. Zhou, Y., Pang, S., Cheng, J., Sun, Y., Wu, Y., Zhao, L., Liu, Y., Lu, Z., Yang, W., Feng, Q.: Unsupervised deformable medical image registration via pyramidal residual deformation fields estimation. arXiv preprint arXiv:2004.07624 (2020)