Completely Contactless and Online Finger Knuckle Identification

Zhenyu ZHOU

July 9, 2022

1 Abstract

2 Introduction

Biometrics is the use of the human body's inherent physiological and behavioral characteristics for person identification. For the physiological characteristics, there are many human characteristics such as retina, iris, face, fingerprints and palm prints; as far as the behavioral characteristics, the available features include gait recognition, voice and other behavioral characteristics. These biometric characteristics are very popular areas of research, attracting many researchers, and have been applied in a wide range of industries. Among human physiological characteristics, finger knuckle, as an optional character, is easier to expose, convenient to collect, and the most importance is that it has been proven to be unique and stable [17]. Therefore, it has also attracted many researchers devoted to it, related works including from the earlier [19], [52] to nowadays [5], [37]. Contactless finger knuckle identification method, a research field of finger knuckle, is more user-friendly, security and more hygienic when compared to contact-based identification. Especially with today's epidemic, contactless identification method can efficiently prevent cross-contamination.

Although the contactless approach offers a lot of convenience, there are two main problems: one is the degradation of matching performance and the other is how to segment the finger knuckle region of interest (ROI) efficiently. Finger knuckle are more prone to deformation in the absence of fixation, which occurs not only in the 2D plane but also in the 3D plane, the presence of variations in ambient light, etc. These factors can cause crease texture of finger knuckle to vary considerably between individuals result in matching performance degradation. In the contactless scenario, finger knuckle are exposed to the interference of a complex background on captured contactless finger knuckle images or videos, and the position, size and number of finger knuckle appearing in each frame varies. This can make automatic finger knuckle detection and segmentation more difficult. If researchers can solve above two main problems, then the development and application of contactless finger knuckle identification will be very promising, and can enhance recognition performance with other biometrics identifiers.

2.1 Related Work

Contactless Finger Knuckle Identification

Many early works on contactless finger knuckle identification, ranging from coding-based methods, subspace methods and texture analysis methods to 3D shape patterns based on 3D image reconstruction, and different methods have been used to achieve highly accurate recognition results. For the traditional recognition algorithms, there are generally two main types: one is holistic-based, and the other is local feature-based. The broad category can be divided into subspace and spectral representation methods for holistic-based [47], [26], [24]. Subspace methods are generally used for data dimensionality reduction and noise reduction [50], such as the use of principal component analysis to reduce the dimensionality of multidimensional data. In contrast to spectral representation methods, image space transformation can be performed as well as image feature enhancement and correlation coefficients for feature extraction [10]. For the processing of local information, there are many algorithms, including, for example, extracting information about the gradient of the image edges, obtaining the boundary points, using other edge extraction algorithms such as Hough change. For example, a 1D log-Gabor filter was used to extract the finger knuckles' features and for the matching phase, the hamming distance was used here for the matching score calculation since it is a local feature-based method [25]. Alternatively, a 2D Gabor filter is used to extract the domain orientation information features of the finger knuckles, and an angular distance calculation is used to calculate the similarity between the different features for the matching score [23]. High recognition accuracy has been achieved for these matching algorithms, even up to 98.67% [47].

Even early work considered application scenario using cell phones for finger knuckle recognition [3], but for finger knuckle segmentation using fixed finger position in the center of the image is not very convenient for the user, for recognition phase log-Gabor is used for feature extraction, and Hamming distance is used for matching. Meanwhile, these paper [18], [5], [14] are the latest in research on the topic.

However, these traditional algorithms have a common problem that these algorithms have to keep changing their corresponding filters and even the corresponding detection parameters under different scenario [53]. In other words, traditional algorithm cannot learn filters and parameters by themselves. With the development of neural network, the generalization ability of feature extraction and feature matching method becomes more robust. Especially, from the oldest LeNet [20] to nowadays EfficientNetV2 [36] model, the CNN models have achieved great success in computer vision tasks. Therefore, the CNN models also can be used on finger knuckle identification tasks. Such as FKNet [5] based on the ResNet-50 [8], it uses the FKNet to classify the subject from the 3D feature of finger knuckle. There are even many that combine traditional methods with CNN models. For example [14], it first extracts different information with Gabor filters, and then uses CNN to learn the most important information.

Contactless Object Detection and Segmentation

For the efficiency of the matching algorithm, the accuracy of the region of interest of object is a factor that can determine the matching efficiency and accuracy. The size of the region of interest should be comparable to the size of the actual target object at the pixel level so that the size obtained after segmentation will not have redundant pixel information, which will reduce the pixel values to be computed for both the extraction and the subsequent matching sessions.

As for the traditional method, their [18], [3] approach is to fix the finger knuckle position in the image when taking the finger knuckle data without complexity background, and then extract the edges of the object. Most importantly, the traditional segmentation algorithm cannot correctly segment the finger knuckles in the presence of complex background interference, multiple finger knuckles in the same field of view, obscured finger knuckles or bent finger knuckles. It is difficult to use traditional object segmentation methods to automatically segment finger knuckles for applications such as in the wild. Models of neural network models for object detection have achieved great success, whether it is the sliding window detection algorithm, the 2-stage series of R-CNN models [7], [6], [30], or the 1-stage YOLO series [29], [27], [28], [2] and SSD models [21] up to the current position, and even the anchor-free based object detection algorithm [46] as well. Each of these models has its advantages. For the 2-stage model, the object detection accuracy is guaranteed, the 1-stage based model is a speedup based on the positive accuracy, and the anchor-free is a further improvement in the detection speed. In this paper, the latest version of the YOLO model series, YOLOv5 [43], is used as the network model for finger knuckle detection because the YOLO series is famous for its fast detection speed and high accuracy.

Limitations and Challenges

2.2 Our Works

As for the contactless finger knuckle identification, the most difficult problem is how to deal with the deformable crease texture of finger knuckle while with different angle. Although many methods have obtained good recognition accuracy, the finger knuckles are easily deformed under practical application scenarios, and the finger knuckle features will change accordingly. Thus, the matching accuracy will be degraded. Because of the above problems, there are corresponding studies to solve the finger knuckle deformation problem and provide new data sets and new methods [18]. The paper [18] first matches on two images for a selected fixed number $32 * 32$ of point pairs for coping with the deformation problem and then uses local feature descriptors on each point pair for matching. And at the RFNet [22], it shifts the feature maps for getting the minimal matching score to solving the palmprint shift problem. Meanwhile, on the person re-identification problem, the paper [1] uses the cross-input neighborhood differences module to calculate difference of one feature maps with another with more range to add robustness to positional differences, and the paper [34] use the normalized correlation layer to achieve the similar problem. However, both of them just shift or calculate with more range but still on the horizontal and vertical direction, no one think about the rotation. Therefore, based on the Soft-Shifted Triplet Loss [22], we can also rotate to deal with more complex deformations.

Put a comparison table as here. The table should list these methods, used database, whether in complex background, matching accuracy (EER and FAR), and so on.

Another problem is tow to automatically segment the finger knuckle region on the real time on the contactless and online finger knuckle identification. From the above contactless finger knuckle paper, they just use the traditional method to fix the finger knuckle position on an image or video. But just like I said before, the traditional method cannot solve the complexity phenomenon. And I have found that on the finger knuckle identification, it seems that no one use deep learning to detection and segmentation. A new finger knuckle object detection algorithm is used, which can automatically extract the region of interest of finger knuckle based on the YOLOv5 [43] model framework and integrates the angle prediction function. It is beneficial to the matching speed and accuracy of the matching algorithm and the automatic target segmentation using the object detection model to extract the ROI region of the major finger knuckle. The angle information of the finger knuckle can be obtained. If the algorithm of ROI segmentation is accurate enough and the accuracy of the rotation is also high enough, this will naturally improve the detection efficiency.

In conclusion, if we want to perform contactless finger knuckle recognition in a real-world scenario or an online scenario, the most critical problem comes from two aspects:how to match with high accuracy in real-world applications, and the second one is how to efficiently perform finger knuckle segmentation and correction. Therefore, our contribution can be summarized as below:

- We design a new loss function for solving finger knuckle deformation with rotation and translation, called RITL loss function. After comparison, our RITL can increase matching perform when compare to STTL. Even, our RITL not only can be used on the finger knuckle identification, but also can be used on other biometric identification.
- We use the YOLOv5-CSL to automatically detection and segmentation. Based on the YOLOv5, we add the angle prediction on the horizontal bounding box for getting better quality segmented ROI, and we can use the angle information to normalize picture. From the segmented finger knuckle by YOLOv5-CSL, the matching perform can be increased when use these finger knuckle database offered.
- We design a cross-platform online finger knuckle identification software for completely contactless and online finger knuckle identification. The system can detect and match finger knuckle patterns from finger images acquired under complex background, using an ordinary smartphone or general laptop camera.

Section 3 will explain our designed RITL loss function and prove that it can be differentiable by equation. Section 4 contains all experiments and corresponding results, including within database experiments and cross database experiments, and 2D finger knuckle and 3D finger knuckle. And also contain the ablation study with changing translation size and rotation size. Section 5 introduces the finger knuckle detection model and how to implement the detection of finger knuckle angle information. Section 6 is to prove the online finger knuckle identification performance.

3 Matching Contactless Finger Knuckle

We choose the Residual Feature Network (RFNet) [22] as our feature extraction backbone, because the model not only is lightweight enough, but it achieves state-of-the-art performance on the palmprint dataset. Meanwhile, the paper [22] uses the soft-shifted triplet loss function, called SSTL to train the model and matching two features for dealing with palmprint shifting problem. As for the triplet loss function [32], it has been a great success in the field of biometrics recognition.

However, in generally, the contactless finger knuckle of the same subject will not only just shift, but also will have local deformable transformation with rotation which is a common problem in the contactless biometrics identification. For solving it, we propose a new loss and also a new matching method. With our proposal loss function, the feature extraction backbone can learn the most robust features that can be rotation invariant, because our loss will get the minimal MSE between two feature maps after translating along the x-axis and y-axis, and rotating clockwise and counterclockwise with hyperparameter. In other words, we can get the minimum value regardless of how the features are rotated, therefore, we call our new loss function rotation invariant triplet loss function (RITL).

3.1 Rotation Invariant Triplet Loss Function

As for a new loss function, the most important point is whether it can be differentiable. With a differentiable loss, the back propagation process can proceed smoothly, and the learnable parameters can be updated to get the minimal loss. In this section, we will discuss the derivation of the RITL loss function. Because our neural networks were trained using the architecture of triplet network [32], we used RITL as loss function to update convolutional kernel of our models.

In generally, the RITL is still a variant of triple loss, so that the RITL can be written as a format of triple loss function as the Equation 1. As for the N, it means the batch size during training iteration, and $T(I^a)$ is the output template of input anchor image I^a through neural network. The hard margin parameter m can determine the distance between different class cluster by pushing them away during training process.

$$RITL = \frac{1}{N}\sum_{i}^{N}[L(T(I_i^a), T(I_i^p)) - L(T(I_i^a), T(I_i^n)) + m]_+ \tag{1}$$

In order to adapt to tasks with different degrees of deformation, and balance performance and speed, we set translation and rotation ranges as a hyperparameter. The $L(T_1, T_2)$ will get the minimal distance of two templates $D_{w,h,\theta}(T_1, T_2)$ after translation and rotation in the range $-W \le w \le W, -H \le h \le H, -\Theta \le \theta \le \Theta$, called minimal translation and rotation distance (MTRD). Meanwhile, the distance $D_{w,h,\theta}(T_1, T_2)$ calculates the pixel-wise MSE value when template T_1 is translated w pixel along x-axis and h pixel along y-axis and rotated θ angle in the Equation 3.

$$L(T_1, T_2) = \underset{-W \le w \le W, -H \le h \le H, -\Theta \le \theta \le \Theta}{min} D_{w,h,\theta}(T_1, T_2) \tag{2}$$

$$D_{w,h,\theta}(T_1, T_2) = \frac{1}{|C_{w,h,\theta}|} \sum_{(x,y) \in C_{w,h,\theta}} (T_1^{(w,h,\theta)}[x, y] - T_2[x, y])^2 \tag{3}$$

Draw a more detail picture with rotation angle, shift size.

In terms of $C_{w,h,\theta}$, it represents the common region between two templates after one template shifted along x-axis with w, shifted along y-axis with h, and rotated with θ, as showed in the Figure 1. As for the (T_a, T_p) pair, we can assume when the T_a is rotated angle of θ_{ap} and shifted with (w_{ap}, h_{ap}) pixels can get the minimal $D_{w_{ap},h_{ap},\theta_{ap}}(T_a, T_p)$, then $L(T_a, T_p) = D_{w_{ap},h_{ap},\theta_{ap}}(T_a, T_p)$. Meanwhile, with the $(w_{an}, h_{an}, \theta_{an})$, the (T_a, T_n) pair can get the minimal $D_{w_{an},h_{an},\theta_{an}}(T_a, T_m)$.

$$\frac{\partial Loss}{\partial T_i^p} = \begin{cases} 0, if(x, y) \notin C_{w_{ap},h_{ap},\theta_{ap}} \; or \; Loss = 0 \\ \frac{-2(T_i^a[[x_{wap}, y_{hap}] * M(\theta_{ap})] - T_i^p[x,y])}{N|C_{w_{ap},h_{ap},\theta_{ap}}|}, otherwise \end{cases} \tag{4}$$

The $M(\theta_{ap})$ is the rotation matrix.

$$\frac{\partial Loss}{\partial T_i^n} = \begin{cases} 0, if(x, y) \notin C_{w_{an},h_{an},\theta_{an}} \; or \; Loss = 0 \\ \frac{-2(T_i^a[[x_{wan}, y_{han}] * M(\theta_{an})] - T_i^n[x,y])}{N|C_{w_{an},h_{an},a_{an}}|}, otherwise \end{cases} \tag{5}$$

As for the $T_i^a[x, y]$ derivation, because we shift and rotate the anchor in the above formula, we

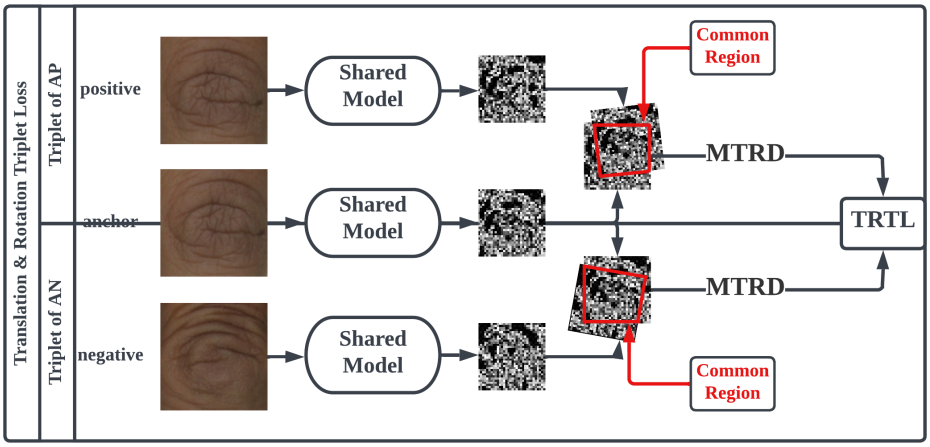


Figure 1: An overview of how to use our translation and rotation triplet loss function (RITL) to train the triplet neural network. We will use the minimal translation and rotation distance (MTRD) to calculate the similarity of two output templates, the common region is the red box after shifting and rotating. During matching process instead of training process, we also use the MTRD to calculate matching scores.

can inversely shift and rotate the positive and negative input feature.

$$\frac{\partial Loss}{\partial T_i^a[x,y]} = -\frac{\partial Loss}{\partial T_i^p[[x-w_{ap}, y-h_{ap}] * M(-\theta_{ap})]} + \frac{\partial Loss}{\partial T_i^n[[x-w_{an}, y-h_{an}] * M(-\theta_{an})]} \tag{6}$$

4 Experiments and Results

We choose the baseline model is the RFNet [22], its performance can outperform DenseNet-BC [12], CompCode [15], DoN [53], Ordinal Code [35], and RLOC [13] algorithms on the palmprint verification problem. For proving TRTL loss function performance, we will compare its performance with Soft-Shift Triplet (STTL)[22] loss function on different public finger knuckle database based on the RFNet [22]. With TRTL loss function, the RFNet is represented by RFNet-TRTL, on the country, RFNet-STTL represents with STTL loss function. Compare to convolution layer or dilated convolution [49], the deformable convolution [54] can solve local deformable by sampling different location and different weight. We also replace the RFNet convolution layer with deformable convolution layer called DeConvRFNet. As for the RFNet and DeConvRFNet, we will firstly pretrain on the HKPolyU Finger Knuckle Images Database (V1.0) [39] as the pretrained weights.

Meanwhile, we will also compare with the FKNet [5] which get the state-of-the-art performance on 3D finger knuckle identification, and EfficientNetV2-S [36]. FKNet performance on the 3D finger knuckle database, 2D finger knuckle and even palmprint database can over SGD [4], CR_L1_DALM, CR_L2 [51], ResNet-50 [8], VGG-16 [33], AlexNet [16], DenseNet-121 [12], and SE-ResNet-50 [11]. Both of FKNet and EfficientNetV2-S are classification neural network. As a classification neural network, it commonly has a problem when the number of classes of testing dataset is not as same as the training set classes, result in fine-tuning on the testing set. Therefore, we use the vector before soft-max layer as the feature vector, and then calculate the MSE of two feature vectors as the similarity score during matching finger knuckle. We use the ResNet-50 pretrained weights as the FKNet initial weights, and use the pretrained weights on the ImageNet21K as the initial weights of EfficientNetV2-S.

We also want to show the performance of TRTL and SSTL on the EfficientNetV2-S model, therefore we keep the same architecture and just change the FC layer of the head part with convolution layer for fitting TRTL and STTL. The changed EfficientNetV2-S model with TRTL called EfficientNetV2-S-TRTL, and with STTL called EfficientNetV2-S-STTL. As same as the EfficientNetV2-S model, we also use the pretrained model weights on the ImageNet21K dataset. In generally, public finger knuckle database already offer segmented finger knuckle images, but we use our YOLOv5-CSL model to segment finger knuckle as our training and testing data during our experiment.

4.1 Model Complexity Analysis

As a completely contactless and online finger knuckle identification, we must choose a model that can meet the requirements of matching speed while ensuring matching accuracy, and even sacrifice matching accuracy for a certain matching speed. We have listed learnable weights of each model, and the corresponding feature extraction time and matching time on the Table 1.

Model	Prams (M)	Input Size	Template Size	FLOPs (B)	Feature Extraction (s)	Matching (s)
DeConvRFNet-STTL	0.36M	128x128	32x32	1.29B	0.02880s	0.01320s
DeConvRFNet-TRTL	0.36M	128x128	32x32	1.29B	0.02880s	0.02516s
EfficientNetV2-S [36]	20.18M	300x300	$1 * N$	5.40B	0.07501s	0.00002s
EfficientNetV2-S-STTL	20.00M	300x300	9x9	5.38B	0.07135s	0.00911s
EfficientNetV2-S-TRTL	20.00M	300x300	9x9	5.38B	0.07135s	0.02581s
FKNet [5]	7.28M	96x64	$1 * N$	0.28B	0.04492s	0.00002s
RFNet-STTL [22]	0.46M	128x128	32x32	1.39B	0.01957s	0.01104s
RFNet-TRTL	0.46M	128x128	32x32	1.39B	0.01957s	0.01984s

Table 1: Comparison time and space complexity of different neural network. Time complexity is the average time on the Ubuntu 22.04 with GeForce RTX 2080 GPU and I7-7800X CPU. When the template size is $1 * N$, it means the training set classes number. For the STTL, the shift size is 4; for the TRTL, the shift size is 4, and add rotation with 4 angles.

......

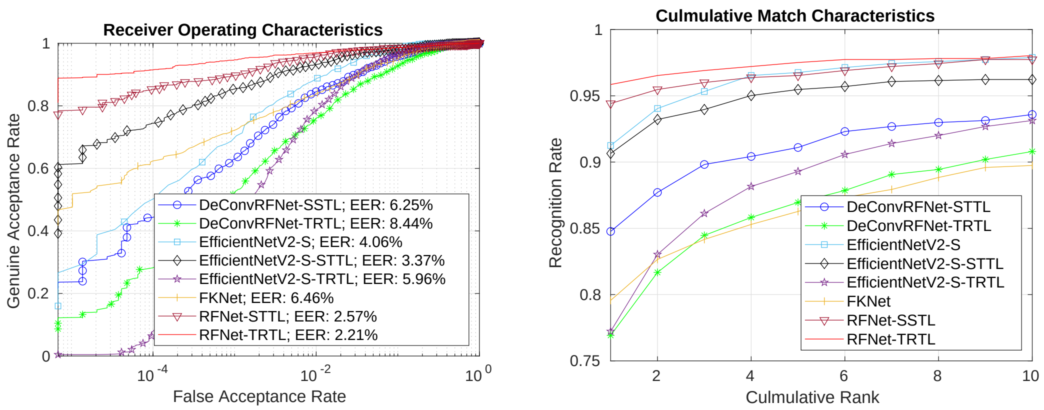
4.2 Within Database Performance Evaluation

4.2.1 Contactless Finger Knuckle Image Database (Version 3.0)

The finger knuckle database [40] can offer contactless finger knuckle of 221 subjects, but only 104 subjects have second session samples. For each session, each subject can offer 6 samples. It is worth mentioning that the finger knuckle sample provided by this database is more challenging and closer to real world scenarios, because the finger knuckle will bend from 0 to 90 degree result in crease deformation.

One-Session Protocol

As for the one-session protocol, I firstly fine-tuned models on the second session 104 subjects dataset, totally $104 * 6 = 624$ images as the testing set. Then use the first session 221 subjects as the testing set result in $221 * 6 = 1326$ genuine matching scores and $221 * 220 * 6 = 291720$ imposter matching scores. From the Figure 2, we can easily find the RFNet is the best model not only on the ROC but also on the CMC. In terms of the baseline model RFNet, our loss function TRTL can improve the matching accuracy when compare to the STTL loss function. Although the finger knuckle of the database with deformation while bend from 0 to 90 degree, the EER of the RFNet-TRTL can arrive at 2.21%. And as top-2 ranking, the RFNet-TRTL recognition rate is about 0.97 on the CMC. As for the rest model, EfficientNetV2-S model performance is better than FKNet and DeConvRFNet. From the performance result, if we just change the convolution layer with deformable convolution, it cannot overcome finger knuckle deformation, even the performance is dropped.

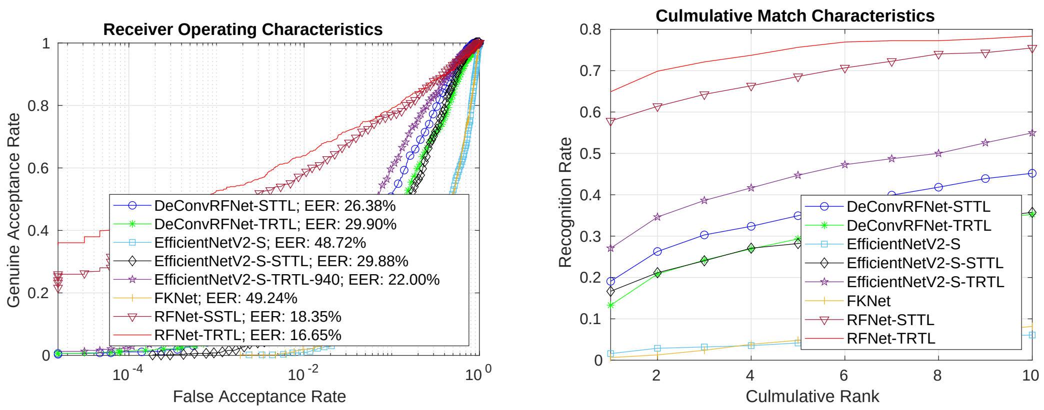


(a) (b)

Figure 2: Comparative ROC (a) and corresponding CMC (b) for one-session on the contactless finger knuckle image database [40].

Two-Session Protocol

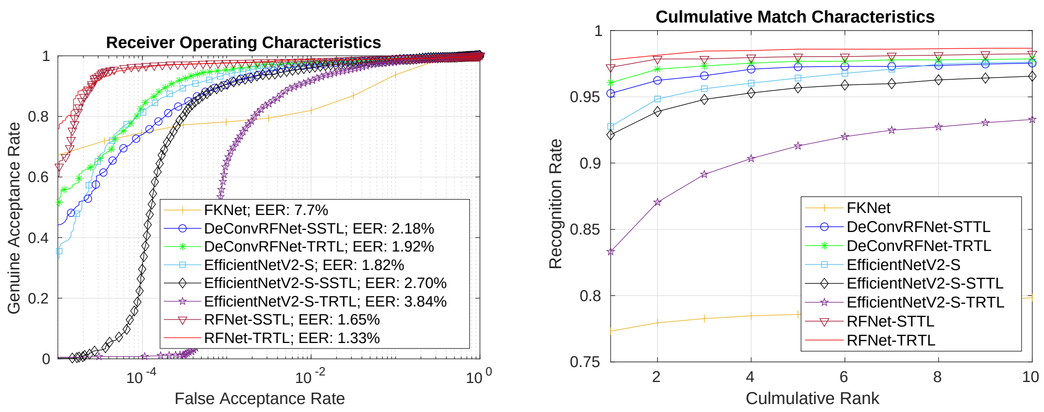
We fine-tune models on the first session subjects who don't provide second session samples, and use two-session protocol to evaluate my model performance on the first session subjects who can offer two-session data. In totally, it will generate $104 * 6 = 624$ genuine scores, and $104 * 103 * 6$ imposter scores. Just like said before, the FKNet and EfficientNetV2-S are classification networks, we use output feature vector to calculate MSE as the matching score. Because the degree of deformation vary on the two-session data, the verification and identification scenarios is more complexity than one-session protocol. Due to these factors, the accuracy on the two-session protocol is much lower than the one-session protocol. However, the RFNet is still the best model, even its EER is half of the EER of other models. Meanwhile, our TRTL loss function still work better than the STTL loss function, with 16.65% and 18.35% respectively on the ROC. As for the CMC, when the cumulative rank value is 2, recognition rate of RFNet-TRTL can arrive at 0.7. From the ROC and CMC Figure 3, we can also get that the STTL and TRTL triplet loss function are better than classification task, because the FKNet and EfficientNetV2-S have the lowest accuracy.



(a) (b)

Figure 3: Comparative ROC (a) and corresponding CMC (b) for two-session on the contactless finger knuckle image database [40].

4.2.2 Index Finger Knuckle of Contactless Hand Dorsal Image Database



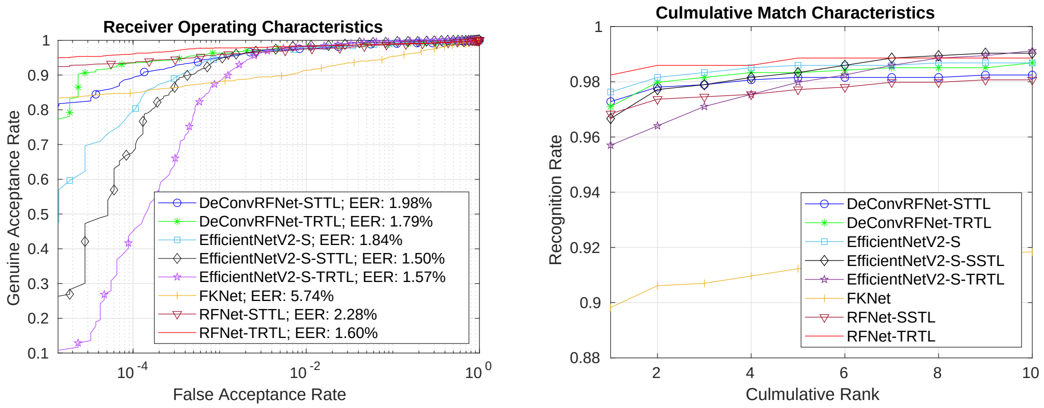
(a) (b)

Figure 4: Comparative ROC (a) and corresponding CMC (b) for one-session on the contactless hand dorsal image database [41].

As for the experiment, the dataset [41] totally contains 712 subjects, and each subject have 5 finger knuckle samples. And we fine-tuned our models on the first sample of each subject, and then use the rest four sample as the testing dataset. For protocol on the database, we use protocol as same as protocol of the FKNet [5]. At the evaluation process, it has $712 * 4 = 2848$ genuine matching scores, and has $712 * 711 * 4 = 2024928$ imposter matching scores. The performance of RFNet-TRTL and RFNet-STTL is similar, but the RFNet-TRTL is slightly better than RFNet-STTL depend on the EER value on ROC. And on the CMC, the RFNet-TRTL still get the best accuracy. We can notice that the FKNet get the worst result when compare to other models. The EfficientNetV2-S model is still better than the FKNet, because EfficientNetV2-S is deeper than FKNet with MBConv block. MBConv block is more robust than the original residual block.

4.2.3 2D Forefinger of 3D Finger Knuckle Database

The HKPolyU 3D Finger Knuckle Images Database [38] can offer reliable 3D finger knuckle pattern (surface normal vector, depth, or curvature) from 2D finger knuckle images, therefore we use its 2D images as our evaluation database. 190 subjects of the database have two-session finger knuckle samples, and 38 subjects offer one-session images. In this kind of situation, two-session protocol is not fit on the database, then we use one-session protocol to evaluate performance. We use the first session 190 subjects images to fine-tune models and then to test on the second session 190 subjects. It has $190 * 6 = 1140$ genuine matching scores and $190 * 189 * 6 = 215460$ imposter matching scores. From the ROC and CMC, we can get a conclusion that the performance of RFNet, DeConvRFNet, and EfficientNetV2-S are similar. However, the FKNet is still the worst one, which EER is 5.74% and the CMC is lower than others. The unchanged thing is that the RFNet with TRTL loss still get the best performance with 1.60% EER, even for the recognition rate on the CMC.



(a) (b)

Figure 5: Comparative ROC (a) and corresponding CMC (b) for one-session on the 3D finger knuckle database[38].

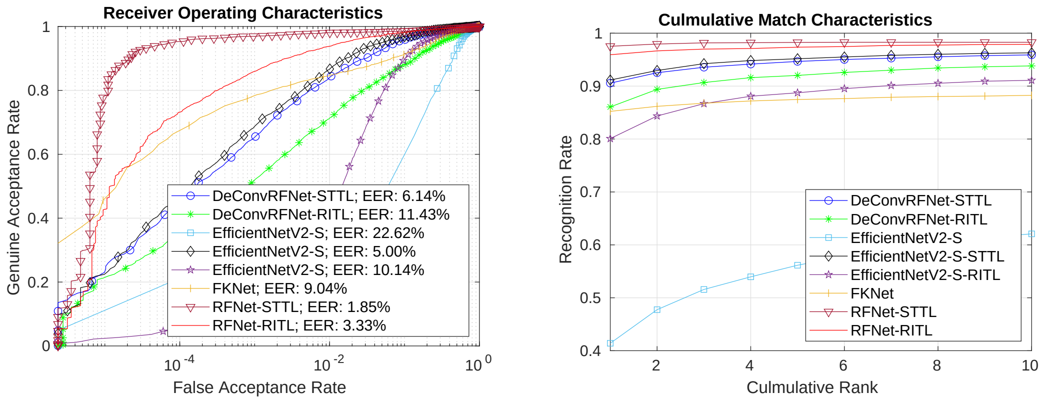
4.3 Cross Database Performance Evaluation

From the within database experiment, we can clearly get a conclusion that the RITL loss function can increase the performance compare to the STTL loss function, and the RFNet is better than the DeConvRFNet, EfficientNetV2-S, and FKNet. Meanwhile, the FKNet performance is the worst one. In this section, we will compare these models' performance on the cross database experiment. For these cross database experiment, it can get the generalization ability of neural network, because these data can be regard as unseen data.

As for the cross database experiment, I firstly pre-trained our models on the Finger Knuckle Images Database V1, and then fine-tuned models on the Finger Knuckle Images Database V3 (with deformation). In the next step, we use our models to test all the finger knuckle of the Hand Dorsal Images Database and the Tsinghua Finger Vein and Finger Dorsal Texture Database (THU-FVFDT3) [42]. Although the THU-FVFDT3 database can offer two-session samples with interval several seconds, but strictly speaking, it is not two-session database. Therefore, I just use the training set of the database (THU-FVFDT-FDT3_Train) as our evaluation dataset.

4.3.1 Hand Dorsal Images Database

Index Finger Knuckle and Middle Finger Knuckle

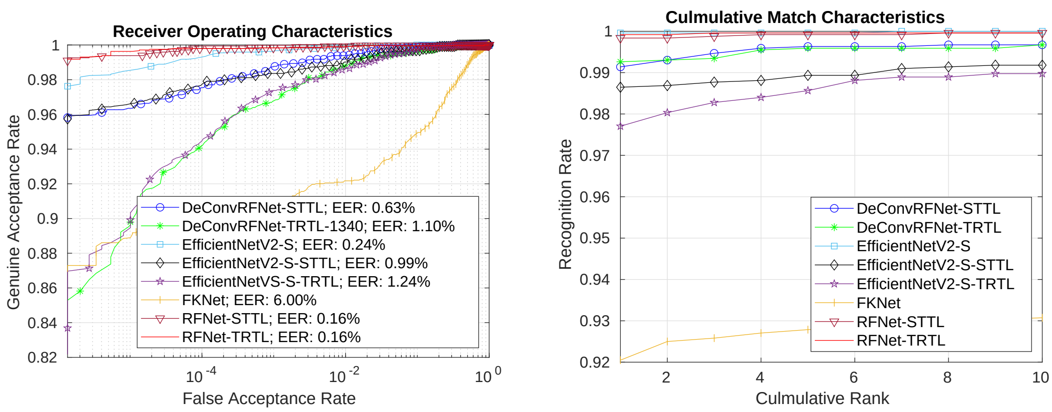


(a) (b)

Figure 6: Comparative ROC (a) and corresponding CMC (b) for one-session of the index finger knuckle on the contactless hand dorsal image database [41].

The database totally has 712 subjects, and each subject has 5 samples of hand dorsal image. Therefore, it will have $712 * 5 = 3560$ genuine matching scores and $712 * 711 * 5 = 2531160$ imposter matching scores for index and middle finger knuckle. Figure 6 is the performance result on the index finger.From Figure 6, all models' cross database performance is similar on the database regardless which finger. We should also notice that STTL is better than RITL on the cross database experiment, while within database, the RITL is better than STTL. It shows that the generalization ability of RITL is not better than STTL to some extent. However, the RFNet-STTL outperform the rest models depend on the ROC and CMC. Even better than FKNet and EfficientNetV2-S, both of them are classification models.

4.3.2 Tsinghua Finger Vein and Finger Dorsal Texture Database



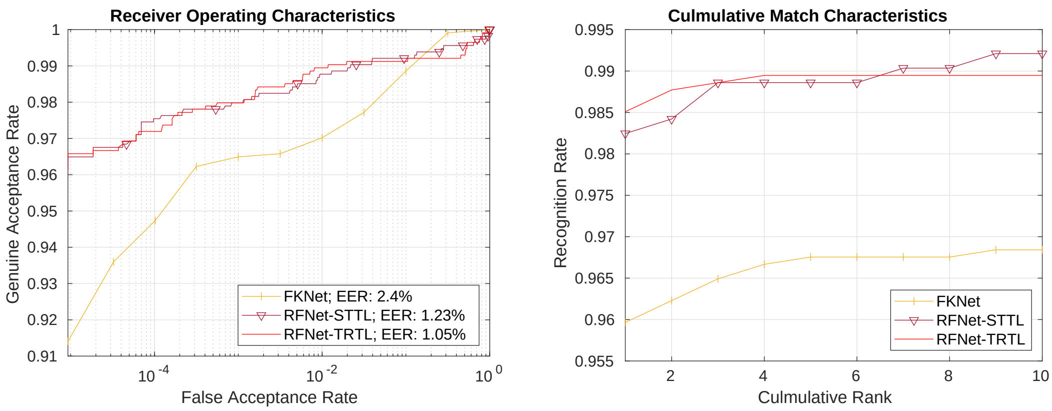
(a) (b)

Figure 7: Comparative ROC (a) and corresponding CMC (b) for one-session of the finger dorsal texture images [42].

The database [42] has 610 subjects, and each subject can offer 4 samples. From the finger dorsal texture images, we can use our YOLOv5-CSL model to segment finger knuckle images as our testing set. Then as the cross database experiment, it will have $610*4 = 2440$ genuine matching scores and $610*609*4 = 1485960$ imposter matching scores. In this database, all models can get very high matching performance from the Figure 7, even the worst FKNet can arrive at 6.00% EER on the database. The RFNet with RITL and STTL get the same accuracy, in terms of the CMC, the recognition rate almost arrive at 100%.

4.4 3D Finger Knuckle Images Database

The 3D finger knuckle images database [38] can offer robust 3D information which can be invariant to changed illuminations, for example, the depth information of the crease of finger knuckle. With the 3D finger knuckle database, the FKNet is the state-of-the-art. Meanwhile, RFNet with TRTL loss function can get the best performance on the within database experiments and cross database experiments when compare to the FKNet on the 2D finger knuckle database. Therefore, we compare the RFNet with FKNet on the database to show the identification performance on 3D finger knuckle database. As for the protocol, it will generate $190*6 = 1140$ genuine matching scores, and $190*189*6 = 215,460$ imposter matching scores from matching matrix. From the Figure 8, RFNet-TRTL still can get the best performance for finger knuckle verification and identification. Form the ROC curve, the EER of the RFNet-TRTL can increase to 1.05% while the EER of the FKNet is 2.4%. Not only on the 2D finger knuckle database, but also on the 3D finger knuckle database, the RFNet-TRTL can outperform the state-of-the-art results.



(a) (b)

Figure 8: Comparative ROC (a) and corresponding CMC (b) for one-session of the 3D finger knuckle database [38].

4.5 Discussion

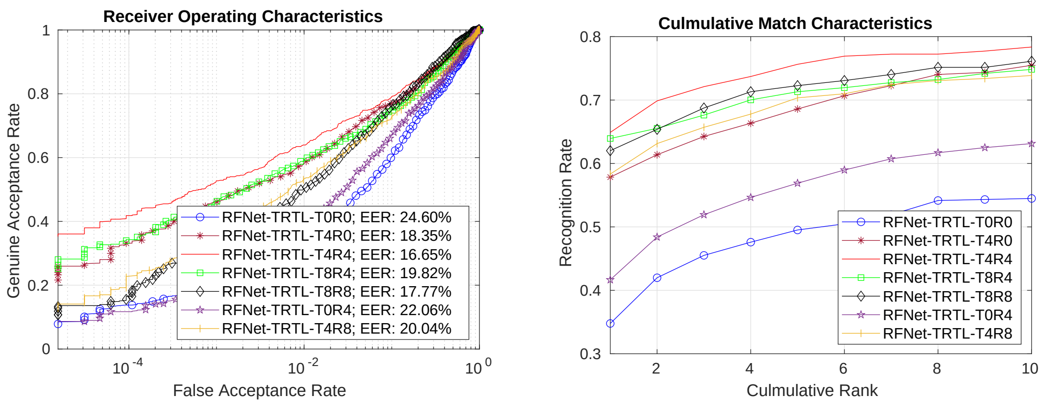
We have compared the identification performance of RFNet with EfficientNetV2-S, DeConvRFNet, and FKNet on the 2D and 3D finger knuckle database, and on the within database and cross database experiments. Due to deeply learned residual features of the RFNet which have already outperformed the state-of-the-art results on the palmprint, it still can get the best performance on the finger knuckle crease when compare to the rest models from above experiment results. Because finger knuckle is very prone to flexing, causing crease texture distortion, if just shift the template, it cannot solve the deformation problem with rotation. Therefore, we design our new TRTL to further solve the problem. With our TRTL loss function when train RFNet and MTRD when matching finger knuckle, the RFNet can increase matching accuracy regardless on the ROC and CMC based on the STTL. Especially on the Finger Knuckle Images Database (Version 3.0) which offer bending finger knuckle with complexity deformation, RFNet-TRTL improved performance is relatively more compare to other database from the Figure 2 and Figure 3. Two-session protocol is more complexity because of changing finger knuckle crease and more complexity deformation when matching process. Form the Figure 3 (a) ROC and (b) CMC, our TRTL with RFNet get the best matching and recognition performance. Meanwhile, our TRTL not only work on the RFNet, but also can work on the DeConvRFNet from the Figure 4 and Figure 5, with TRTL loss function, the matching and recognition performance also can increase.

From these experiment results, we can also get a conclusion is that EfficientV2-S model is better than FKNet from the within database experiment, and even on the Tsinghua finger knuckle database. EfficientNetV2 model can outperform the ResNet on the ImageNet [31], in other words, the EfficientNetV2 model can extract robust feature than ResNet on the ImageNet. Because EfficientNetV2 replace the residual block with inverted residual block, and use MBConv as a block unit. As for MBConv block, it uses depth-wise convolutions to decrease training weights and use Squeeze-Excited block as channel attention. Meanwhile, the depth of EfficientNetV2-S is deeper than the FKNet. On the contrary, the FKNet use the ResNet-50 fist conv3 as the feature extract model. EfficientNetV2 use the more light, advance and efficient module than ResNet.

However, TRTL generalization ability is lower than STTL loss from the cross database experiment, except on the Tsinghua database. On the cross hand dorsal database, Figure 6, EER of RFNet with TRTL performance will drop from about 2.0% to 5.0%. As for the rest model, performance with TRTL also drop with corresponding value when compare to STTL. But in the within database experiment, these model with TRTL loss is better than STTL loss. It shows our TRTL can affect the back propagation during training process to the different model weights, in other words. And another phenomenon is that EfficientNetV2-S with SSTL and TRTL cannot work. From the Table 1, if the input image size is 300x300, EfficientNetV2-S with STTL and TRTL will output 9x9 template size. The output feature size is too small when use the STTL and TRTL loss function, inversely, the performance will drop while translation and rotation.

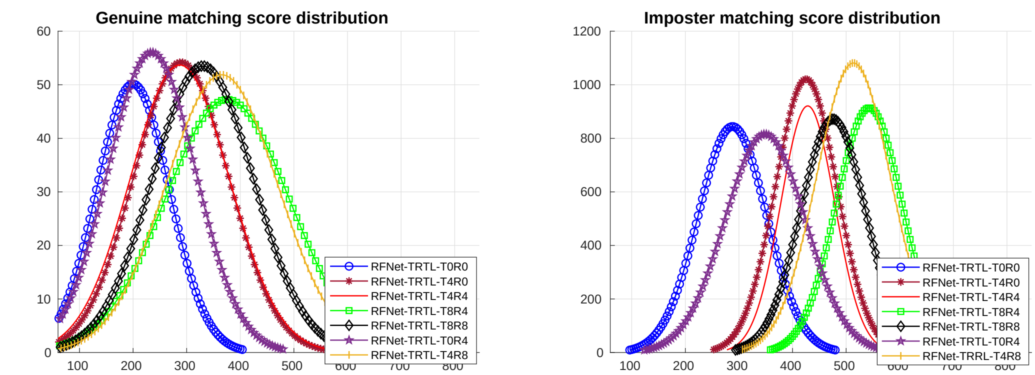
4.6 Ablation Study

One of the ablation study is change the shift and rotation hyperparameter to show the performance with the two-session protocol on the deformable finger knuckle database [40]. We change the shift size from 0 to 8, and the rotation angle also from 0 to 8, and the interval value is 4. From the Figure 9, we can get from the ROC and CMC



(a) (b)

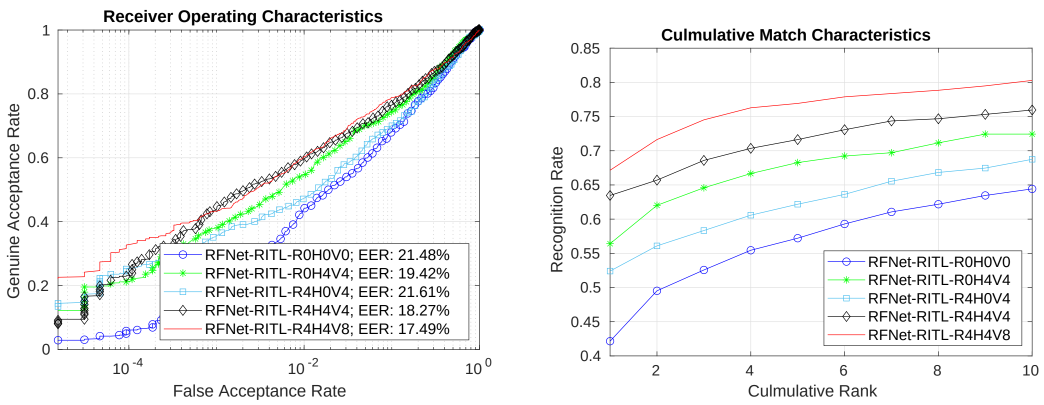
Figure 9: Comparative ROC (a) and corresponding CMC (b) for two-session of the Finger Knuckle Database (Version 3.0) [40]. For approving our TRTL loss function efficiency, we change the translation parameter and rotation parameter to show the different matching performance.



(a) (b)

Figure 10: Genuine matching score distribution (a) and Imposter matching score (b) for two-session of the Finger Knuckle Database (Version 3.0) [40]. For approving our TRTL loss function efficiency, we change the translation parameter and rotation parameter to show the different matching performance.

The other ablation study is change the input image size. From segmented finger knuckle by YOLOv5x-CSL, the minimal size of ROI is about $189 * 224$. Therefore, we keep the same ratio to change all segmented finger knuckle to $184 * 208$ due to rectangle bounding boxes. And in this kind of situation, we change the vertical and horizontal size with different size. And we make the vertical shifting size bigger than horizontal shifting size, get the result in the Figure 11.



(a) (b)

Figure 11

Put the Grad-CAM to show neural network output feature from the original input picture. ....

5 Online Contactless Finger Knuckle Identification

With TRTL loss, the RFNet [22] can outperform state-of-the-art methods. In the previous section, we have estimated its verification and identification performance on different public finger knuckle database, including within-db and cross-db experiments. As for a completely contactless and online finger knuckle identification, the finger knuckle detector is a very important module for automatically detect and segment finger knuckle region. However, as for traditional segmentation algorithm, they cannot correctly segment the finger knuckles in the presence of complex background interference, multiple finger knuckles in the same field of view, obscured finger knuckles or bent finger knuckles. Meanwhile, as for neural network, the current based on YOLO [29], [27], [28], [2], [43] and R-CNN [7], [6], [30], [9] series object detection and segmentation approaches cannot simultaneously obtain the angle of finger knuckle and the segmentation with high precision. Especially, the angle of the finger knuckle is a vital factor for identification. If we can get the angle of finger knuckle, we can use angle information to align two feature maps for increasing matching accuracy and efficiency. For solving above problems, we propose rotated bounding box detection based on YOLOv5 model for segmenting and getting angle information.

5.1 Contactless Finger Knuckle Detection

5.1.1 Oriented Bounding Box Based on YOLOv5

In order to solve the problem of finger knuckle detection in the real world, we choose to use YOLOv5 model because the YOLO series is famous for its fast detection speed and high accuracy. Especially, the YOLOv5's [43] speed can meet our online detection requirements.

Oriented Bounding Box

However, the YOLOv5 just detect horizontal bounding boxes which cannot offer angle information and will segment a lot of background information. In order to solve these above problem, a rotated bounding box will be predicted instead of horizontal bounding box. As analyzed in this paper [48], the rotated bounding boxes loss will mainly come from angular periodicity and the exchangeability of edges. When use the long side definition of rotated bounding box, it can deal with the exchangeability of edges problem. Meanwhile, using classification task to predict angle can make model easier to train. A periodic coding method called Circular Smooth Label (CSL) [48] soft coding can also solve the problem that One-Hot cannot distinguish class relationship. Formula 7 $g(x)$ is the window function to smooth One-Hot label, and r is a window function of the radius.

$$CSL(x) = \begin{cases} g(x), & \theta - r < x < r + \theta \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

Furthermore, in this paper, we used the Gaussian function for the Equation 7 window function, a commonly available function, and used a window radius of 6 to smooth the labels.

Loss function

The original YOLOv5 loss function can have three components. The formula can be simply written as $Loss = CIOU\_Loss + Loss_{conf} + Loss_{class}$. Since the rotated bounding box is based on the modification of YOLOv5, only the angle classification loss is added more. So the total loss function is as expressed in Equation 8, with the addition of $Loss_{angle}$ to YOLOv5 loss function.

$$Loss = CIOU\_Loss + Loss_{conf} + Loss_{class} + Loss_{angle} \tag{8}$$

$$\begin{aligned} Loss_{angle} = \sum_{i=0}^{S^2} I_{ij}^{obj} \sum_{a\in[0,180)} & [\hat{P_i(a)} log(P_i(a)) + \\ & (1 - \hat{P_i(a)}) log(1 - P_i(a))] \end{aligned} \tag{9}$$

5.1.2 Contactless Finger Knuckle Dataset

Our task is to detect finger knuckles in the contactless and online scenario, but by understanding current public finger knuckle database, their data are collected at specific conditions such as certain angle, certain light. In this kind of situation, this kind of data cannot represent real images of finger knuckle in real world. In order to address the shortcomings of current public finger knuckle dataset for contactless detection, we use a web crawler to get images from the Unsplash [44] where the keywords are finger knuckles. The Unsplash is an image site that offers uploads and downloads, and uses a copyright license that allows users to download and use them for free or even for commercial use [45]. We have downloaded 2347 images, there are 738 images without knuckles, and these images can be used as background training, and the rest 1609 images that contain at least one finger knuckle are the positive samples for the network model. In the network training process, we use crawled images, 169 finger knuckle images from the HKPolyU Finger Knuckle Database (V1.0) [39], and 64 finger knuckles images from the HKPolyU Hand Dorsal Database [41] as for the training set. And we use the rest data as testing set to evaluate performance. The most important part is the data augmentation which contains flip, rotation, resize, translate and mosaic.

5.1.3 Contactless Finger Knuckle Detection

Detection Performance

The YOLOv4, YOLOv5x, and YOLOv5m model predict horizontal bounding box, while the remaining YOLOv5 model predict oriented bounding box with CSL classification, called YOLOv5-CSL. We can see the performance difference between these variations of the YOLOv5 model from the Table 2. Among the downloaded 2580 images, 100 images were randomly selected as the testing set. Compare the performance of the horizontal bounding box prediction and oriented bounding box.

Model	Total Time/ms (1024x1024)	Number of Layers	mAP^{val} 0.5	AP of Major FK	AP of Minor FK
YOLOv5x-CSL	40.9	407	**89.9**	**89.6**	**90.1**
YOLOv5m-CSL	23.3	263	85.7	88.9	80.4
YOLOv5x	33.3 ms	407	87.3	86.5	88.0
YOLOv5m	12.1ms	263	84.8	84.5	85.1

Table 2: Comparison of the accuracy of the different models of the YOLO series for detecting finger knuckle. The calculated values of mAP were measured at a detection threshold of 0.001 as well as an IOU threshold of 0.5. All the experiments are test on the GTX 2080 GPU and I7-7800X CPU, and the total time includes the inference and NMS process.

Segmentation Performance

This section aim to compare quality of finger knuckle between YOLOv5-CSL segmented and dataset offered. Because the segmented finger knuckle on the 3D Finger Knuckle Dataset already have high quality, I mainly test on the Index Finger Knuckle of Hand Dorsal Dataset and the Finger Knuckle Dataset V3 (with deformable).

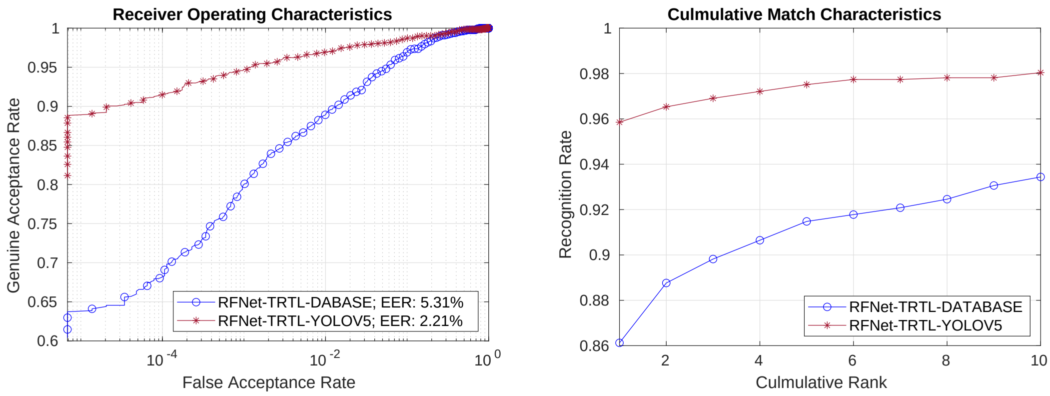


Figure 12: Compare performance on the Finger Knuckle V3 Dataset (with deformable)

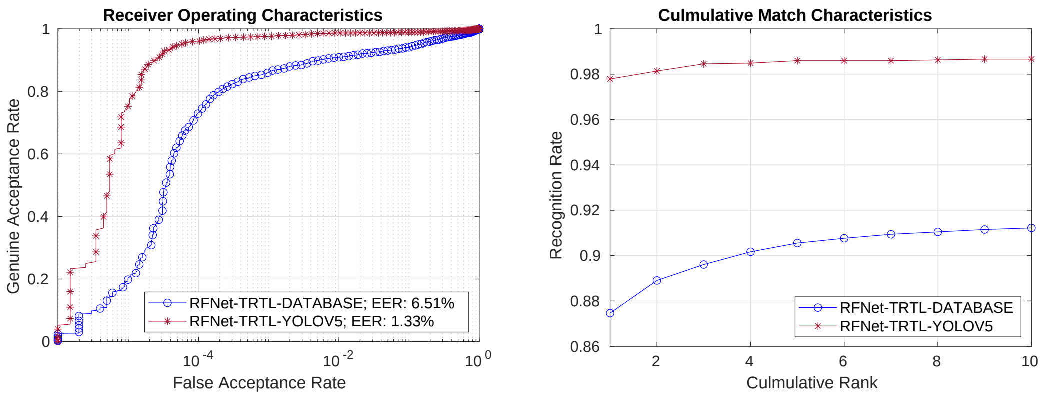
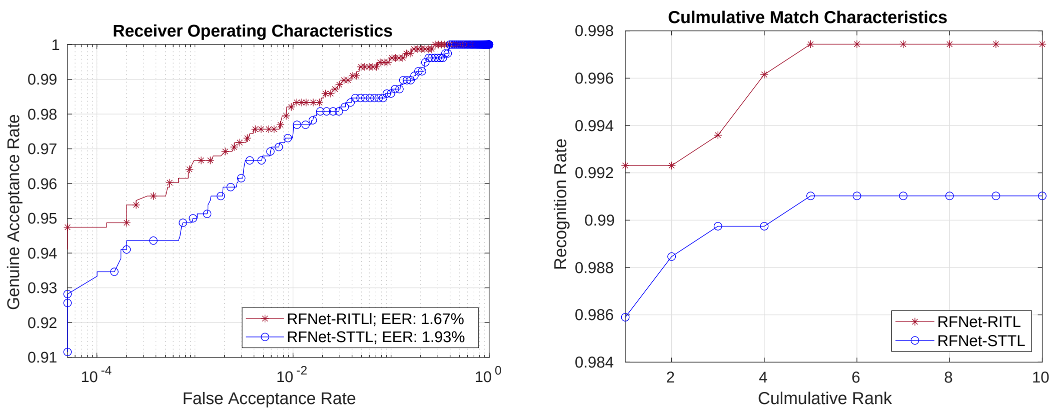


Figure 13: Compare performance on the Index Finger of the Hand Dorsal Image Database.

From the above figures, we can clearly get the conclusion that quality of segmented finger knuckle of YOLOv5 is better than the segmented finger knuckle of dataset through the ROC curve and CMC curve. Especially on the Hand Dorsal Image Database, the EER value can drop from 6.51% to 1.33%.

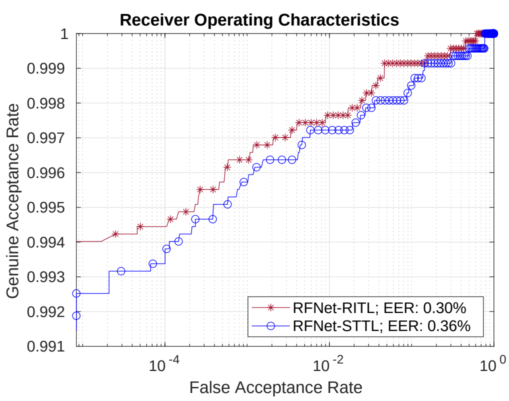
5.2 Online Contactless Finger Knuckle Identification Performance

For proving our contactless and online finger knuckle identification performance, we capture 52 subjects and each subject can offer about 15-20s contactless finger knuckle video. The minimum frame rate of the videos we offer is 30 frames per second. We choose two method to get the contactless finger knuckle images, one is that we get 1 image per second, another one is that we get 6 images per second for testing online matching performance. Because the shortest video is 15s, for 1 image per second, we can totally get $52 * 15 = 780$ finger knuckle images for keeping the same number of samples result in $52 * 15 = 1789$ genuine matching scores and $52 * 51 * 15 = 39780$ imposter matching scores. In terms of the 6 images per second, we can get $52 * 15 * 6 = 4680$ finger knuckle samples, result in $52 * 90 = 4680$ genuine matching scores and $52 * 51 * 90 = 238680$ imposter matching scores. ....



(a) (b)

Figure 14: Comparative ROC (a) and corresponding CMC (b) with one session protocol for



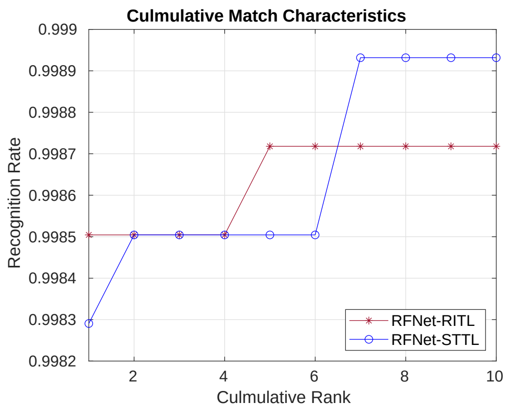


Figure 15

6 Conclusions and Future Work

7 References

[1] Ejaz Ahmed, Michael Jones, and Tim K Marks. "An improved deep learning architecture for person re-identification". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 3908–3916.

[2] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. "Yolov4: Optimal speed and accuracy of object detection". In: *arXiv preprint arXiv:2004.10934* (2020).

[3] KamYuen Cheng and Ajay Kumar. "Contactless finger knuckle identification using smartphones". In: *2012 BIOSIG-Proceedings of the International Conference of Biometrics Special Interest Group (BIOSIG)*. IEEE. 2012, pp. 1–6.

[4] Kevin HM Cheng and Ajay Kumar. "Contactless biometric identification using 3D finger knuckle patterns". In: *IEEE transactions on pattern analysis and machine intelligence* 42.8 (2019), pp. 1868–1883.

[5] Kevin HM Cheng and Ajay Kumar. "Deep feature collaboration for challenging 3D finger knuckle identification". In: *IEEE Transactions on Information Forensics and Security* 16 (2020), pp. 1158–1173.

[6] Ross Girshick. "Fast r-cnn". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1440–1448.

[7] Ross Girshick et al. "Rich feature hierarchies for accurate object detection and semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 580–587.

[8] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[9] Kaiming He et al. "Mask r-cnn". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2961–2969.

[10] Pablo Hennings, Marios Savvides, and BVK Vijaya Kumar. "Verification of biometric palmprint patterns using optimal trade-off filter classifiers". In: *International Conference Image Analysis and Recognition*. Springer. 2005, pp. 1081–1088.

[11] Jie Hu, Li Shen, and Gang Sun. “Squeeze-and-excitation networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7132–7141.

[12] Gao Huang et al. “Densely connected convolutional networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 4700–4708.

[13] Wei Jia, De-Shuang Huang, and David Zhang. “Palmprint verification based on robust line orientation code”. In: *Pattern Recognition* 41.5 (2008), pp. 1504–1513.

[14] Rajiv Kapoor et al. “Completely Contactless Finger-Knuckle Recognition using Gabor Initialized Siamese Network”. In: *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*. IEEE. 2020, pp. 867–872.

[15] AW-K Kong and David Zhang. “Competitive coding scheme for palmprint verification”. In: *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.* Vol. 1. IEEE. 2004, pp. 520–523.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems* 25 (2012).

[17] Ajay Kumar. “Importance of being unique from finger dorsal patterns: Exploring minor finger knuckle patterns in verifying human identities”. In: *IEEE transactions on information forensics and security* 9.8 (2014), pp. 1288–1298.

[18] Ajay Kumar. “Toward pose invariant and completely contactless finger knuckle recognition”. In: *IEEE Transactions on Biometrics, Behavior, and Identity Science* 1.3 (2019), pp. 201–209.

[19] Ajay Kumar and Ch Ravikanth. “Personal authentication using finger knuckle surface”. In: *IEEE Transactions on Information Forensics and Security* 4.1 (2009), pp. 98–110.

[20] Yann LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[21] Wei Liu et al. “Ssd: Single shot multibox detector”. In: *European conference on computer vision*. Springer. 2016, pp. 21–37.

[22] Yang Liu and Ajay Kumar. “Contactless palmprint identification using deeply learned residual features”. In: *IEEE Transactions on Biometrics, Behavior, and Identity Science* 2.2 (2020), pp. 172–181.

[23] Rajiv Mehrotra, Kameswara Rao Namuduri, and Nagarajan Ranganathan. “Gabor filter-based edge detection”. In: *Pattern recognition* 25.12 (1992), pp. 1479–1494.

[24] Abdallah Meraoumia, Salim Chitroub, and Ahmed Bouridane. “Fusion of finger-knuckle-print and palmprint for an efficient multi-biometric system of person recognition”. In: *2011 IEEE International Conference on Communications (ICC)*. IEEE. 2011, pp. 1–5.

[25] Abdallah Meraoumia, Salim Chitroub, and Ahmed Bouridane. “Personal Recognition by Finger-Knuckle-Print Based on Log-Gabor Filter Response”. In: ().

[26] Shubhangi Neware, Kamal Mehta, and AS Zadgaonkar. “Finger knuckle identification using principal component analysis and nearest mean classifier”. In: *International Journal of Computer Applications* 70.9 (2013).

[27] Joseph Redmon and Ali Farhadi. “YOLO9000: better, faster, stronger”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 7263–7271.

[28] Joseph Redmon and Ali Farhadi. “Yolov3: An incremental improvement”. In: *arXiv preprint arXiv:1804.02767* (2018).

[29] Joseph Redmon et al. “You only look once: Unified, real-time object detection”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 779–788.

[30] Shaoqing Ren et al. “Faster r-cnn: Towards real-time object detection with region proposal networks”. In: *Advances in neural information processing systems* 28 (2015), pp. 91–99.

[31] Olga Russakovsky et al. “Imagenet large scale visual recognition challenge”. In: *International journal of computer vision* 115.3 (2015), pp. 211–252.

[32] Florian Schroff, Dmitry Kalenichenko, and James Philbin. “Facenet: A unified embedding for face recognition and clustering”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 815–823.

[33] Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).

[34] Arulkumar Subramaniam, Moitreya Chatterjee, and Anurag Mittal. “Deep neural networks with inexact matching for person re-identification”. In: *Advances in neural information processing systems* 29 (2016).

[35] Zhenan Sun et al. “Ordinal palmprint represention for personal identification [represention read representation]”. In: *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR’05)*. Vol. 1. IEEE. 2005, pp. 279–284.

[36] Mingxing Tan and Quoc Le. “Efficientnetv2: Smaller models and faster training”. In: *International Conference on Machine Learning*. PMLR. 2021, pp. 10096–10106.

[37] Ahmad S Tarawneh et al. “DeepKnuckle: Deep Learning for Finger Knuckle Print Recognition”. In: *Electronics* 11.4 (2022), p. 513.

[38] *The HKPolyU 3D Finger Knuckle Images Database:* `https://www4.comp.polyu.edu.hk/~csajaykr/3DKnuckle.htm`.

[39] *The HKPolyU Contactless Finger Knuckle Images Database (V-1.0):* `http://www4.comp.polyu.edu.hk/~csajaykr/fn1.htm`.

[40] *The HKPolyU Contactless Finger Knuckle Images Database (Version 3.0) :* `https://www4.comp.polyu.edu.hk/~csajaykr/fn2.htm`.

[41] *The HKPolyU Contactless Hand Dorsal Images Database:* `http://www4.comp.polyu.edu.hk/~csajaykr/knuckleV2.htm`.

[42] *Tsinghua University Finger Vein and Finger Dorsal Texture Database (THU-FVFDT3):* `https://www.sigs.tsinghua.edu.cn/labs/vipl/thu-fvfdt.html`.

[43] Ultralytics. *YOLOv5*. `https://github.com/ultralytics/yolov5`. 18 May 2020.

[44] Unsplash. *Unsplash*. `https://unsplash.com/`.

[45] Unsplash. *Unsplash.com*. ”Unsplash License”. Retrieved 11 January 2017.

[46] Ying Xin et al. “PAFNet: An Efficient Anchor-Free Object Detector Guidance”. In: *arXiv preprint arXiv:2104.13534* (2021).

[47] Wankou Yang, Changyin Sun, and Zhenyu Wang. “Finger-knuckle-print recognition using Gabor feature and MMDA”. In: *Frontiers of Electrical and Electronic Engineering in China* 6.2 (2011), pp. 374–380.

[48] Xue Yang and Junchi Yan. "Arbitrary-oriented object detection with circular smooth label". In: *European Conference on Computer Vision*. Springer. 2020, pp. 677–694.

[49] Fisher Yu and Vladlen Koltun. "Multiscale context aggregation by dilated convolutions". In: *arXiv preprint arXiv:1511.07122* (2015).

[50] David Zhang, Xiaoyuan Jing, and Jian Yang. *Biometric image discrimination technologies*. IGI Global, 2006.

[51] Lin Zhang et al. "3D palmprint identification using block-wise features and collaborative representation". In: *IEEE transactions on pattern analysis and machine intelligence* 37.8 (2014), pp. 1730–1736.

[52] Lin Zhang et al. "Online finger-knuckle-print verification for personal authentication". In: *Pattern recognition* 43.7 (2010), pp. 2560–2571.

[53] Qian Zheng, Ajay Kumar, and Gang Pan. "A 3D feature descriptor recovered from a single 2D palmprint image". In: *IEEE transactions on pattern analysis and machine intelligence* 38.6 (2016), pp. 1272–1279.

[54] Xizhou Zhu et al. "Deformable convnets v2: More deformable, better results". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 9308–9316.