IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2021.Doi Number

Deep Learning-based Vein Localization on Embedded System

Chaoying Tang, Member, IEEE, Shuhang Xia, Mengen Qian, and Biao Wang, Member, IEEE
College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, 211106 China

Corresponding author: Biao Wang (wangbiao@nuaa.edu.cn).

This work was supported by The Key Research & Development Programs of Jiangsu Province (BE2018720) and The Open project of Engineering Center of Ministry of Education (NJ2020004).

ABSTRACT Venipuncture is a common process in medical treatment. In the fight against pandemic like COVID-19, it is often very difficult for medical staff to carry out venipuncture accurately, since the staff have to wear safety glasses and surgical gloves. In this work, we designed an embedded system which implements deep learning algorithm to localize veins from color skin images. The proposed method consists of a fully convolutional neural network (CNN) as encoder and feature extractor, a dilated convolution module, and a transposed convolution module as decoder. A synchronized RGB/Near Infrared (NIR) image database was constructed to provide the mapping information between the two image fields. A combined loss function which includes a per-pixel loss and a perceptual loss was presented to optimize the network parameters. To make the model adaptive to different images, a histogram specification scheme was adopted to transform the color style of an image. The model was then implemented on a NVIDIA Jetson TX2 development kit. Comprehensive experiments were conducted on different databases to evaluate the proposed method and the embedded system. Experimental results showed that the system has satisfactory performance and a promising perspective in daily medical treatment.

INDEX TERMS vein localization, convolutional neural network, NVIDIA Jetson TX2

I. INTRODUCTION

Beginning in December 2019, a series of pneumonia cases of unknown causes were reported in Wuhan, China. Deep sequencing analysis from lower respiratory tract samples revealed a novel coronavirus, which was named COVID-19 later [1]. Soon after that, the virus has caused a large scale of epidemic and spread to more than 215 countries around the world. Up to now a total of 77,307,971 confirmed cases and 1,701,085 deaths have been reported globally [2]. Huge number of patients rushed to hospitals in a very short time [3]. To protect themselves from the highly contagious virus, medical staff had to wear protective clothing, face mask, safety glasses and surgical gloves, sometimes even multiple pairs of surgical gloves. The protective equipment became a burden to the medical staff. In addition, glasses usually fog up because the mask redirects the warm breath upward instead of forward, which forms droplets on the lenses, adding much more difficulties to their work, such as venipuncture, as shown in Fig. 1.

Venipuncture is one of the most important procedures for both medical testing and treatment, when a cannula is inserted into a vein on the back of the hand or forearm through a needle to get blood sample or inject saline or medicine [4]. According to statistics, about 90% of patients require venipuncture for the intravenous route of treatment and more than one billion venipunctures are being performed as a basic requirement for most diagnostic tests every year [5]. In this process, medical staff usually look for veins by observation and touch, which puts higher requirements on the skills and experience of practitioners. Even without safety glasses or surgical gloves, it is not an easy task, especially for children and patients with dark skin and obesity [6]. In the United States, more than 400 million venous intubations are performed daily, and the success rate for the first attempt is approximately 72.5% [7]. In the fight against COVID-19, the success rate of venipuncture will be further reduced due to the tension of patients and the obstruction caused by protective clothing and surgical gloves. Multi-person assistance is sometimes required to successfully perform venipuncture.

Fig. 1. Medical staff fighting COVID-19

(a) medical staff wearing double pairs of surgical gloves; (b) safety glasses fog up and form droplets on the lenses

Therefore, the technique of localizing veins is of great help to medical treatment, especially in urgent and large-scale medical cases. It can significantly improve the success rate of the first venipuncture and shorten the puncture time, thus reducing the trauma and pain of the patient [8]. It can also reduce the risk of infection, enable the patient to get treatment faster, win the rescue time, and greatly improve the work efficiency of the medical staff.

Other than vein localization in venipuncture, knowing the distribution characteristics of veins has a specific importance to human health. For example, knowing the appearance of blood vessels in legs is significant for the diagnosis of varicosity and other diseases. Therefore, the collection of vein images also has important application values in the field of medical diagnosis and treatment.

In this paper, a low-power and real-time deep learning-based vein localization method is designed and implemented on an NVIDIA Jetson TX2 development kit. The method learns the mapping from RGB to NIR skin images. To optimize the network parameters, a combined loss function is presented which includes not only a per-pixel loss between the output and the NIR images, but also a perceptual loss based on high-level vein features. The implementation of the proposed method on the embedded system NVIDIA Jetson TX2 makes the task of vein localization small and portable. The paper is organized as follows. Section 2 discusses the related work. Section 3 describes the design of the method and the embedded system. Section 4 shows the experimental results which evaluate the proposed method and system over different image databases. Section 5 concludes the findings.

II. RELATED WORK

Currently, some products of vein locators are available on the market to improve accessibility of veins. They are dependent on ultrasound, NIR or other equipment. There are also some localization methods developed based on image processing algorithms. As an embedded platform, the NVIDIA Jetson TX2 is probably the most suitable choice in terms of design speed and cost effectiveness to develop proof of concept and final solutions to a portable vein localization device. In this section, we will give a short review of the related work.

A. Devices showing vein distribution

Since veins are distributed under the skin, ordinary cameras cannot obtain clear distribution images. In recent years, some instruments for detecting veins have been developed, such as ultrasonic detector [9-11], multispectral cameras and infrared imaging instrument [12-14]. Qu et al. [15] transmitted ultrasonic pulses with high-frequency to the human body. In the process of propagation, some energy was transmitted and the other was reflected back when encountering tissue interfaces with different acoustic impedance. In this way, the time interval between the transmitted and the echo pulses determines the distance between the skin surface and the vein.

Some researchers use multispectral cameras to take skin images at different wavelengths to detect veins. Pavanjeet et al. [16] used multispectral camera to take skin image, and designed near-infrared LED light source with specific wavelength and light intensity which can obtain effective vein images for different skin colors. Shahzad et al. [17] classified human skin into four categories, and used a multispectral camera to find the optimal near-infrared wavelength for each category of skin, so as to maximize the contrast of vein images.

Hemoglobin in blood has the property of absorbing near-infrared or far-infrared rays radiated or reflected by the human body [7]. Serkan and Ö mer et al. [18-19] used near-infrared camera to obtain palm vein images, and used stereo camera to obtain vein depth information, thus improving the clarity of vein images. Ng [20] used far-infrared equipment to obtain the vein images of neck, arm and hand back, and then processed the images with pseudo color through the look-up table to improve the contrast between vein and skin. Ahmed et al. [21] developed an automatic puncture / injection device, which uses near-infrared camera and image processing algorithm to obtain the position of veins. It uses a Bluetooth device to control the automatic injection device, and ensures that the needle is aligned with the vein position through real-time video feedback.

All of the aforementioned instruments need auxiliary equipment or devices, such as infrared light emission and imaging device, ultrasonic emission and reception device, multispectral camera, and so on. These devices increase the volume and weight of the system, and also greatly raise the cost, thus restricting the popularization of vein localization technology. For example, the price of an infrared vein imaging device on the market is as high as $2000. These cause inconveniences to daily medical care.

B. Methods based on image processing algorithms

Some technologies based on image processing were proposed recently to localize veins from RGB images. Tang et al. [22] proposed a vein localization algorithm based on optics and skin biophysics. It models the inverse process of skin color formation in an image and derives the spatial distributions of biophysical parameters, i.e., the percentage of epidermal volume occupied by melanin, the percentage of dermal volume occupied by blood, and the depth of dermis, where vein patterns can be observed. In [23], they further improved the optical model and took the hypodermis into consideration. Reichman equation based on the Schuster-Schartzchild approximation was also employed to replace the K-M model as a more accurate model of radiation transfer. These are called optical methods, and can achieve clear vein patterns in some images. A problem with the methods is that they are pixel-wise algorithms, and no neighboring information was taken into consideration. Therefore, the resultant images are quite noisy.

Song et al. [24] proposed a vein localization method based on RGB images. It used multispectral Wiener estimation to acquire reflectance information from skin, then localized veins from an image taken by a digital camera. A color calibration is necessary in this method for the specific illumination, which affects the localization performance. Watanabe et al. [25] proposed a method that visualizes veins by utilizing the saturation of color information in an image. It can achieve good results on dorsal veins since the skin there is usually very thin, but no experiments on skin of other body parts were reported. Ma et. al. [26] proposed a Generative Adversarial Network (GAN) model, which builds two generators learning from an RGB-NIR dataset for uncovering veins from RGB images. However, the accuracy of the model cannot be guaranteed since its loss function lacked the constraint of vein locations.

Compared with vein localization technologies using auxiliary equipment, the number of algorithms based on image processing remains quite limited.

C. Hardware implementations

During the past decade, deep learning has demonstrated tremendous success in artificial intelligence. In particular, it has been popularly applied in the field of image processing and pattern recognition. Diverse hardware solutions for deep learning techniques have been proposed in recent years. They range from standalone solutions to heterogeneous systems and systems-on-chip (SoC), which include field-programmable gate arrays (FPGAs), application-specific integrated circuits (ASICs), CPUs, and graphics processing units (GPUs). NVIDIA is one of the most important GPU manufacturers and has been dominating the current market with its dedicated GPU programming framework called CUDA. NVIDIA Jetson TX2 is a promising AI SoC powered by NVIDIA Pascal GPU architecture.

Some researchers have used NVIDIA Jetson TX2 to develop hardware implementation for different tasks. Beatriz et al. proposed a real time embedded system for a deep learning-based multiple object visual tracking and mobile edge computing application. The results in terms of power consumption and frame rate demonstrate the feasibility of deep learning algorithms on embedded platforms [27]. Goya et al. designed deep learning methods for diabetic foot ulcer detection and localization. They also developed a mobile device for real-time application on Jetson TX2 [28]. Toan et al. proposed an enhanced detection and recognition system of road markings implemented on Jetson TX 2, which has the benefit of a less processing time [29]. Haut et. al. explored the use of low-power consumption architectures and deep learning algorithms for hyperspectral image classification. They revealed that Jetson TX2 offers a good choice in terms of performance, cost, and energy consumption [30]. To the best of our knowledge, there are not previous deep learning-based solutions for vein localization running on Jetson TX2.

III. METHODOLOGY

Deep learning is a branch of machine learning that models high-level abstractions in data based on a set of optimization algorithms using multiple processing layers with complex structures or multiple non-linear transformations. To localize veins hidden in RGB images, we construct a deep neural network model which automatically extracts features in the image, and learns the mapping relationship between RGB and NIR images from a dataset. The proposed method consists of four parts: dataset preparation, network model construction, model training, and embedded system design.

A. Dataset preparation

A key idea in deep learning method is to learn not only the nonlinear mapping between the inputs and outputs, but also the underlying structure of the data. Our dataset comes from the Forensic Skin Image Databases of Nanyang Technological University in Singapore [31], containing synchronous RGB and NIR inner arm images, as shown in Fig. 2. A 2-CCD multi-spectral prism camera, JAI AD-080-CL, is used to take images from 150 subjects. The camera splits incoming light into two separate channels - a visible color channel from 400 to 700nm and a NIR channel from 750 to 920 nm, and provides simultaneous images of different light spectrums through a single optical path. Since hemoglobin in blood has an especially strong attraction to NIR spectrum, veins are visible in NIR images, which provides prior information about veins.

The training process of deep neural networks is influenced by the size of the input image. More time and memory consumption are necessary for extracting features from large input images. In addition, the background in the original image will affect the convolution operation, and the non-skin pixels will cause unnecessary calculations and reduce the processing efficiency. The noise contained in the background can also adversely affect the learning accuracy. Therefore, we do not use the whole arm image as input. Due to the irregular shape of arm, we use MATLAB to cut a rectangle region containing complete vein patterns from the skin image to form a new image pair. Part of the data set is shown in Fig. 3.

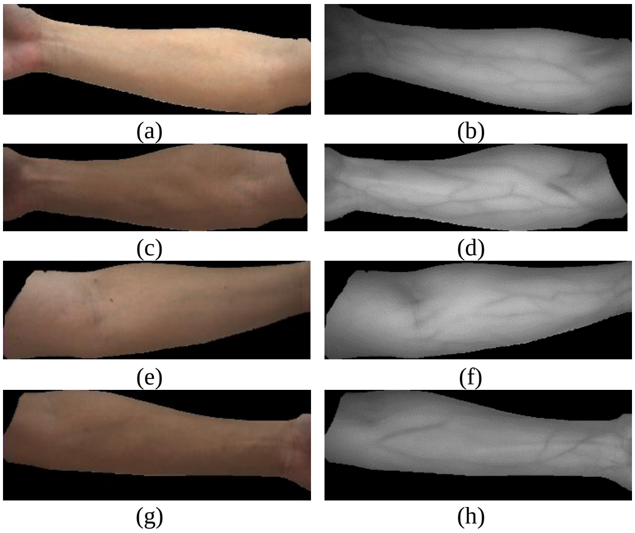


Fig. 2. Synchronous RGB/NIR arm images
(a) and (b), (c) and (d), (e) and (f), (g) and(h) are four pairs of RGB/NIR arm images.

In order to increase the diversity of the training data, the skin images are randomly rotated in the range of [0 °, 180 °] around their centers, and the final image size is 256*256 pixels. We also carry out random transformation on the hue and saturation of the images. The color values are first changed to the range of [0 1]:

$$\begin{aligned} R' &= \frac{R}{255} \\ G' &= \frac{G}{255} \\ B' &= \frac{B}{255} \end{aligned} \tag{1}$$

Then the hue, saturation, and value are calculated according to Eqs. (2)-(4).

$$H = \begin{cases} 0^{\circ}, & \Delta = 0 \\ 60^{\circ} \times \left(\dfrac{G' - B'}{\Delta} + 0 \right), & C\max = R' \\ 60^{\circ} \times \left(\dfrac{B' - R'}{\Delta} + 2 \right), & C\max = G' \\ 60^{\circ} \times \left(\dfrac{R' - G'}{\Delta} + 4 \right), & C\max = B' \end{cases} \tag{2}$$

$$S = \begin{cases} 0 & C\max = 0 \\ \dfrac{\Delta}{C\max} & C\max \neq 0 \end{cases} \tag{3}$$

$$V = C\max \tag{4}$$

where $C\max = \max(R', G', B')$, $C\min = \min(R', G', B')$, and $\Delta = C\max - C\min$. $H \in [0, 360°]$, $S \in [0, 1]$, $V \in [0, 1]$. We give some random modifications to the hue and saturation through Eq.(5).

$$\begin{aligned} H &= H + a \\ S &= S + b \\ V &= V + c \end{aligned} \tag{5}$$

where $a \in [-16°, 16°]$, $b \in [-4/255, 4/255]$, $c \in [-8/255, 8/255]$. Finally, 200 image pairs are obtained for training. Some results are shown in Fig. 4.

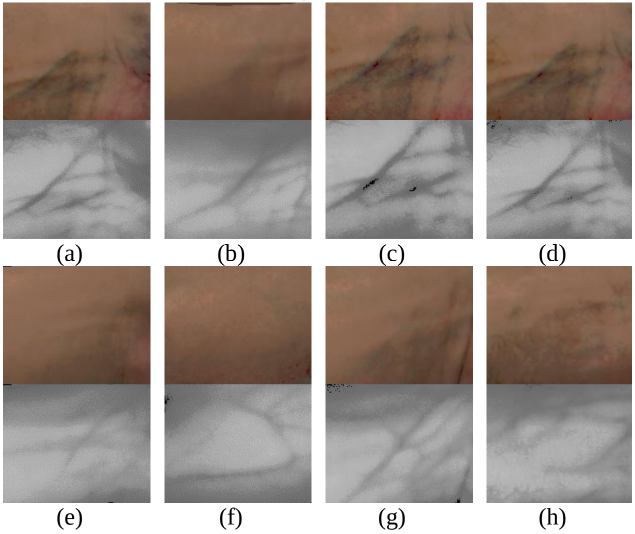


Fig. 3. Image patches extracted for CNN model training
In each example of (a) to (h), the first row is patches cut from the RGB images. The second row is the corresponding NIR image patches.

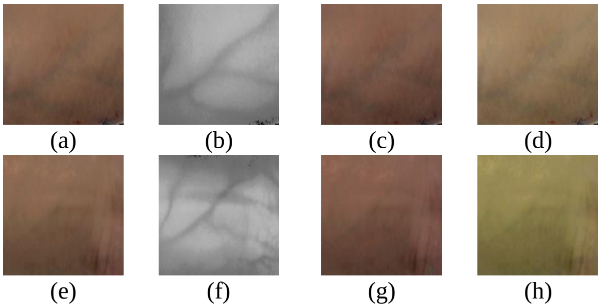


Fig. 4. Image patches after data augmentation
(a) and (b) are a pair of skin patches; (c) and (d) are the results of data augmentation from (a); (e)-(h) are another set of examples.

B. Deep neural network design

Deep neural networks have demonstrated their competences over conventional techniques in the field of image classification and object detection. By extracting features at various levels, they are very good at classifying objects in both non-medical and medical areas. The prerequisite step to classification is localizing objects in an image. From this point of view, the task of vein localization is a specific task of object localization. The proposed method consists of three parts: a fully convolutional network as encoder and feature extractor, a dilated convolution module, and a transposed convolution module as decoder.

To learn a mapping relationship from RGB space to NIR space, a deep convolution network is necessary to extract features from input images. However, when we increase the number of layers, there is a common problem in deep learning called Vanishing/Exploding gradient. This causes the gradient to become 0 or too large, thus the training and test error increases with the number of layers. To solve the problem, we use ResNet34 [32] as encoder and feature extractor. It has skip connections which skip training from a few layers and connects directly to the output. If any layer hurts the performance of architecture, then it will be skipped by regularization.

The features of veins are extracted and serve as input for the later stages. A dilated block [33] is added after the feature extractor module. By increasing the field of view in the convolution, it guarantees the accuracy required by the algorithm and reduces the depth of the network and the complexity of the parameters. The dilated convolution module includes four dilated convolution layers and four Relu layers as activation functions, where each dilated convolution layer is connected to a Relu layer.

Since the target output is an image showing vein information instead of a label in the traditional classification task, a transposed convolution module is set as a decoder to output a vein image after the dilated convolution module. In order to improve the model performance, each layer in the feature extractor is connected to each layer of the decoder separately, and finally the sigmoid function is used in the output layer. The overall framework of the proposed method is depicted in Fig. 5. The size of the input image is 256*256 pixels, and the feature size of each layer through the network model is shown in Table I.

TABLE I
FEATURE DIMENSIONS OF THE PROPOSED NETWORK MODEL

Encoder	Feature dimensions	Decoder	Feature dimensions
Firstconv	[64, 128, 128]	Deconder4	[256, 16, 16]
Maxpool	[64, 64, 64]	Deconder3	[128, 32, 32]
Layer1	[64, 64, 64]	Deconder2	[64, 64, 64]
Layer2	[128, 32, 32]	Deconder1	[64, 128, 128]
Layer3	[256, 16, 16]	ConvTranspose	[16, 256, 256]
Layer4	[512, 8, 8]	Finalconv	[1, 256, 256]

C. Loss Function and Training

Vein localization based on CNN can be framed as an image transformation task, where the input is an RGB skin image and the output image encodes geometric information about veins under the skin. Traditionally, the neural network is trained in a supervised manner, using a per-pixel loss function to measure the difference between the localized result and the NIR image, which is regarded as the ground-truth of vein distribution. However, the per-pixel losses do not capture perceptual differences between the actual and the expected outputs. During training, perceptual losses measure image similarities more robustly than per-pixel losses. In this paper, we propose a combined loss function which includes a per-pixel loss and a perceptual loss, thus fusing the benefits of both.

Unlike natural images with various contents, skin images usually have low frequency components. However, veins in the resultant image have obvious edges. In a digital image, edges are the points where the image brightness changes sharply or has discontinuities. Areas with edges in an image usually have large gradients. For the areas with low frequency component, the change of pixel values is relatively smooth, which corresponds to smaller gradients [34]. Here the gradient values represent the perceptual vein information in a skin image. A horizontal gradient map and a vertical gradient map are extracted from a training NIR image, and also from the resultant image. The mean square errors are then calculated from the corresponding gradient maps. The average of gradient maps in two directions is called a perceptual loss [35]:

$$Loss_{percep} = \mu_x * (g_{xt} - g_x)^2 + \mu_y * \left(g_{yt} - g_y\right)^2 \quad (6)$$

where g_x and g_{xt} are the gradient maps in the horizontal direction of the output image and target NIR image, respectively, g_y and g_{yt} are the gradient maps in the vertical direction of the output image and target NIR image, respectively, and μ_x and μ_y are the weight coefficients in the two directions. Since the majority of veins in arms are horizontally distributed, the gradient in the horizontal direction should be given more importance. In this paper, μ_x and μ_y are set as 0.6 and 0.4, respectively.

The second part of the loss function is the per-pixel loss between the resultant image and the corresponding NIR image**,** which is regarded as the ground truth of vein distribution. The mean squared error (MSE) is used as the loss function:

$$Loss_{pixel} = (x_t - x)^2 \quad (7)$$

where x is the localized result and x_t is the corresponding NIR image.

The total loss function is the weighted sum of the two parts:

$$Loss = (1 - \lambda) * Loss_{pixel} + \lambda * Loss_{percep} \quad (8)$$

where λ is the weight coefficient and is set as 0.3.

Instead of the traditional Stochastic Gradient Descent (SGD), we use a different parameter optimization method, Stochastic Weight Averaging (SWA) [36, 37]. There are two important ingredients that make it work. First, SWA uses a modified learning rate schedule so that instead of simply converging to a single solution, it continues to explore the set of high-performing networks. Second, it averages the weights of the networks traversed by SGD. SWA solutions end up in the center of a wide flat region of loss, while SGD tends to converge to the boundary of the low-loss region, making it susceptible to the shift between train and test error surfaces. Therefore, the training process will converge faster.

D. Embedded system design

For the remote deep learning applications, NVIDIA Jetson TX2 is the latest mobile computer hardware with a GPU card, as shown in Fig. 6. It is not only a single board computer, but also a ready-to-use development kit with size of 5.0×8.7 cm and weight of 85 g. It contains an NVIDIA Pascal GPU 1.3 GHz with 256 CUDA cores, a quad-core 2.0 GHz 64bit

ARMv8 A57 processor, and a dual-core 2.0 GHz superscalar ARMv8 Denver processor. The ARM CPU consists of two ARMv864-bit cores. The CPU cores and the GPU share 8 GB DRAM memory. Its specifications are listed in Table II. Jetson TX2 provides a command line tool for switching operation modes at run time, which adjusts the CPUs and GPU clock speeds by dynamic voltage and frequency scaling [27]. In addition, the board can reduce the calculation time by using the CUDA library to optimize the multiplication and addition operations of deep neural networks.

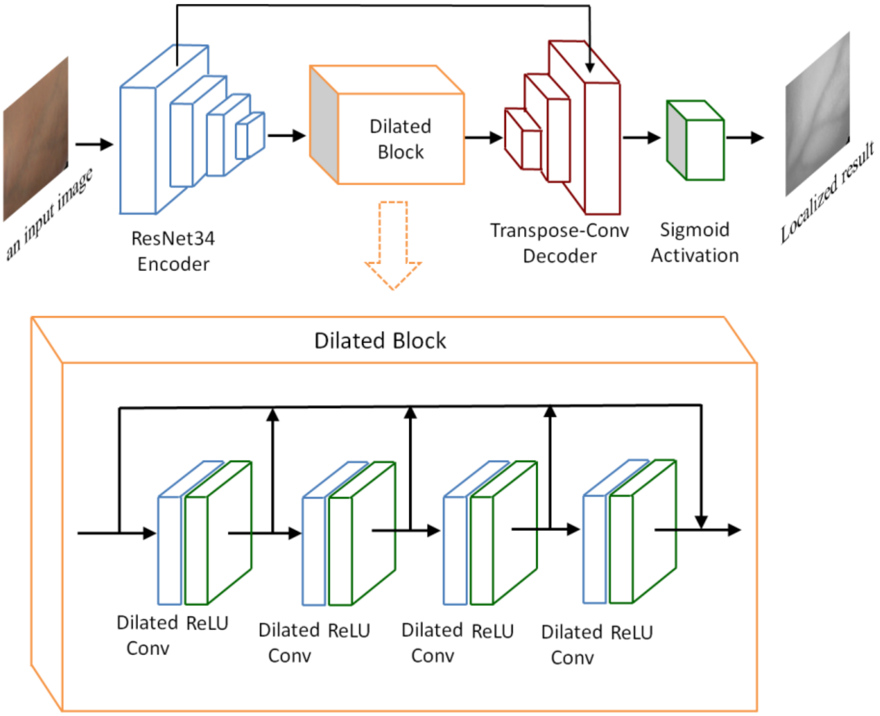


Fig. 5. Illustration of the proposed CNN

Fig. 6. Jetson TX2 embedded system adopted in the proposed method.

NVIDIA provides the jetpack development kit for the Jetson series. We choose the version jetpack4.2.1 for our embedded system. Jetpack includes many API interfaces and deep learning neural network acceleration libraries. We use this development kit to configure the embedded device with ubuntu18.04 operating system and cuda10.0 & cudnn8.0. In addition, for our network model framework, we have configured some packages and libraries such as Pytorch and OpenCV.

Although Jetson TX2 is a compact and portable device that can be used in various locations, it is not capable of training large deep learning models. Therefore, the training is carried out on a GPU server, and then the best model is deployed to Jetson TX2. This process is shown in Fig. 7.

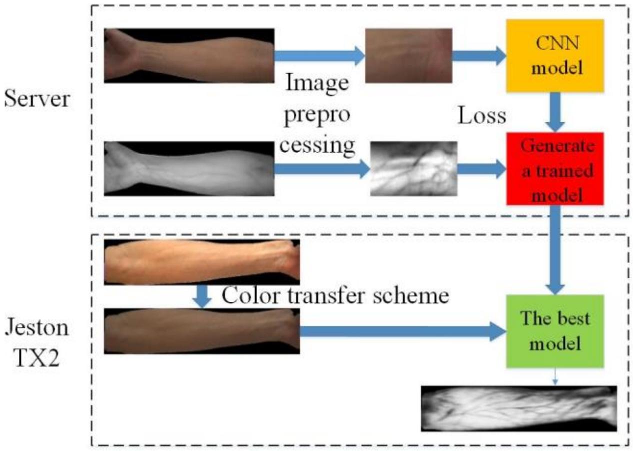


Fig. 7. The process of deploying the model to TX2

TABLE II
SPECIFICATIONS OF JETSON TX2

Item	Specification
Size	50 mm * 87 mm * 10.4 mm (width * height * depth)
GPU	NVIDIA Pascal^{TM}, 256 CUDA cores
CPU	HMP Dual Denver 2(2 MB L2) + Quad ARM^{R} A57 (2 MB)
Operating system	Linux for Tegra R28.1
Memory	8 GB
Data storage	32 GB

E. A histogram specification scheme

We use images taken by the JAI camera to provide training samples for the model. For the reason of different lighting conditions and camera properties, the images taken by other cameras may be very different from the training samples. It may affect the performance of the method. To make it adaptive to different images, we adopt a histogram specification scheme that transforms the color style of an input image so that its histogram matches a specified histogram [38]. An image with classical color style in the training set is selected as the reference which provides the target histogram.

The mapping formula of the original histogram equalization is:

$$s = T(r) = \int_0^r p(t)dt \tag{9}$$

where r is the intensity value of a pixel in the original image, T is the transformation process, s is the pixel value of the transformed image, and p is the probability density function (PDF) of the original image. The PDF of the transformed image is uniformly distributed. For an image with L gray levels, the process can be written in a discrete form:

$$s_t = T(r_t) = \sum_{i=0}^{t} p(r_i) \quad t = 0,1,\cdots,L-1 \tag{10}$$

The histogram specification scheme yields an image with a PDF that follows a specified shape $f(z)$ for $z \in [0,1]$. If the final image is also transformed by histogram equalization, the result would be an image that also has a uniform PDF:

$$s = G(z) = \int_0^z f_z(u)du \tag{11}$$

so the specified histogram can be obtained from Eq.s (9) and (11):

$$z = G^{-1}(s) = G^{-1}[T(r)] \tag{12}$$

For a normalized image with L gray levels, the specification process can be implemented based on the formulation of:

$$s_t = T(r_t) = \sum_{i=0}^{t} f_r(r_i) \quad t = 0,\frac{1}{L-1},\frac{2}{L-1},\cdots,1 \tag{13}$$

$$s_t = G(z_t) = \sum_{j=0}^{t} f_z(z_j) \quad t = 0,\frac{1}{L-1},\frac{2}{L-1},\cdots,1 \tag{14}$$

The red, green, and blue channels of the reference color image are extracted and performed histogram equalization respectively. The cumulative distribution $T(z)$ of each pixel z is obtained. The same procedure is carried out on a test image, and the cumulative distribution $G(s)$ is obtained. The difference between the corresponding channel of reference and test images is calculated, and a gray scale mapping is established accordingly.

IV. EXPERIMENTAL RESULTS

A. Embedded system development and parameter optimization

As mentioned in section III, the image data are augmented by random rotation around their centers and affine transformation on the hue and saturation of the images. After that we obtain 200 pairs of synchronized RGB-NIR arm image patches. 80% of the data is randomly selected as the training set and the other 20% is used as the test set. The training is carried out on Ubuntu 16.04, a 64-bit operating system, using the Pytorch deep learning framework, and implemented on a NVIDIA 1080Ti (12G video memory) GPU, which takes a total of 50 minutes. The network parameters are optimized for 81920 iterations with batch size of 2. The learning rate is initially set as 2 × 10^{-4} and changed in each epoch according to Eq. (15).

$$lr = \begin{cases} x & y < 0.3z \\ \frac{1}{4}x & 0.3z \le y < 0.6z \\ \frac{1}{16}x & 0.6z \le y < 0.9z \\ \frac{1}{64}x & y \ge 0.9z \end{cases} \tag{15}$$

where x is the initial learning rate, y is the current training epoch, and z is the total number of training epochs. The parameter adjustment method is SWA, which can speed up the training of the network and improve the efficiency, as shown in Fig. 8. We use 256 epochs for training the model, which are sufficient for both training and validation. To process a complete image in the test stage, we firstly use the HSV color space separation and the Otsu's method [39] to extract the skin area. Then its color style is modified based on histogram specification. After that, the preprocessed image is input to the deep model. Finally, the output localized result is masked again with only the skin area remained.

We select the best model based on minimum validation loss and deploy it to Jeston TX2. Pytorch specifically designed for ARM64 is installed to produce inference from the vein localization model trained on the GPU server. We compare the localizing results and do not find any difference from those obtained on the server. The results obtained are completely consistent, which means the model has been successfully implanted to the embedded system, as shown in Fig. 9.

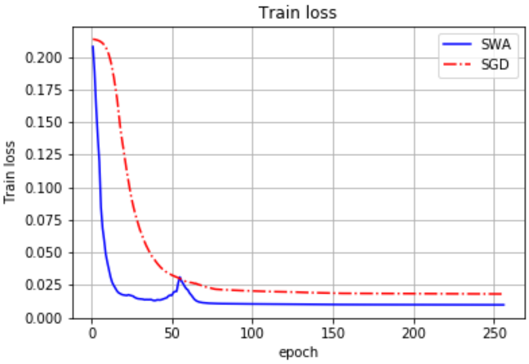


Fig. 8. Illustration of the training process

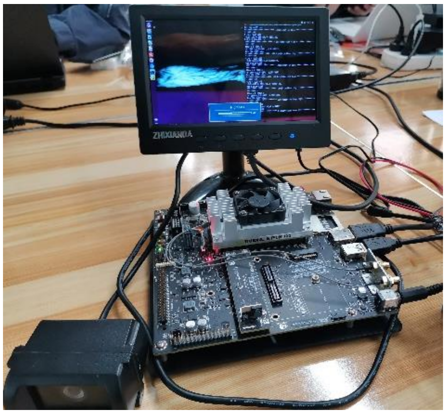
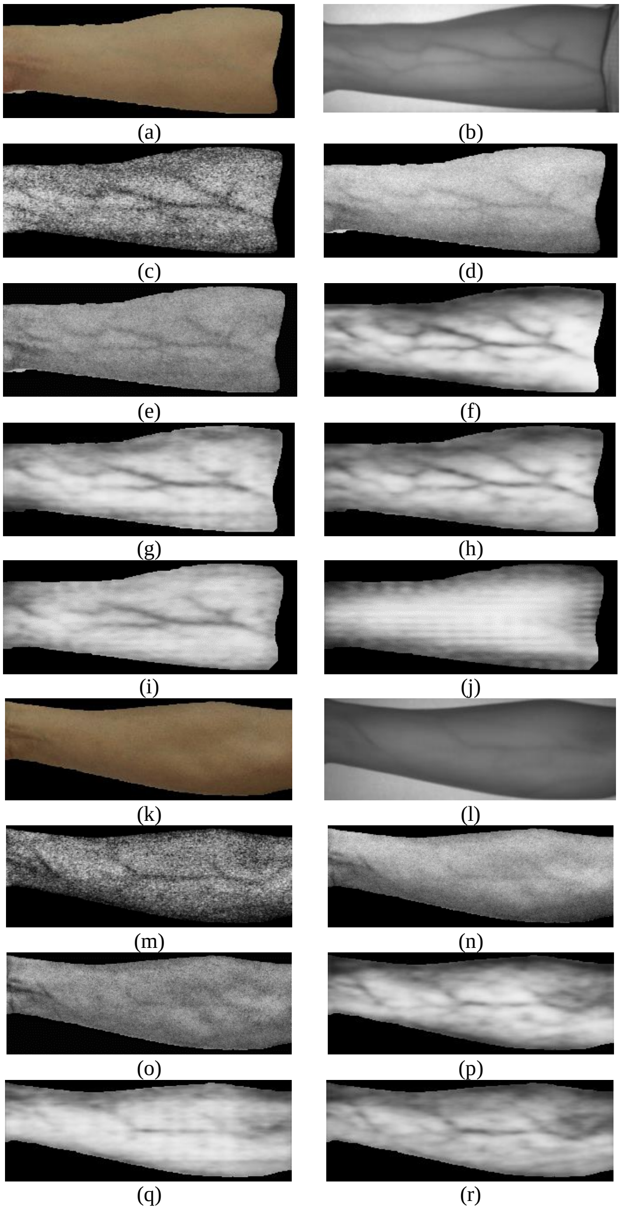


Fig. 9. Illustration of the embedded system

B. Evaluation of the vein localization model

To evaluate the performance of the proposed method, two datasets: JAI image dataset and DSLR image dataset are employed. The first dataset was taken by the JAI camera, i.e., they are from the same camera of the training samples. The second dataset was taken by a DSLR camera, and the model is Cannon 500D. JAI image dataset and DSLR image dataset were constructed at the same time. Specifically, a pair of synchronized RGB/NIR images was taken by the JAI camera, and an RGB image was taken by the DSLR camera from each arm of 150 persons. Only the JAI image dataset has synchronous NIR images, therefore, both subjective and objective evaluations are carried out in this dataset. We compare the proposed method with three state-of-the-art methods for vein localization: Tang et al.'s optical method [23], Song's Wiener method [24], and Watanabe's method [25]. The proposed method uses ResNet34 as encoder and feature extractor. In this experiment, we also compare the deep model with ResNet18, ResNet50, VGG and AlexNet as encoder and feature extractor.

Some experimental results are shown in Fig. 10. Fig. 10(a) is the image of a right inner forearm. Fig. 10(b) is its corresponding NIR image. Figs. 10(c)~(e) are the localized results from the Optical method, the Wiener method, and Watanabe's method, respectively. Fig. 10(f) is the result from the proposed method. It can be seen that the result from the optical method contains a lot of noises since it is a pixel-wise algorithm. The Wiener method and Watanabe's method cannot obtain satisfactory result in some skin areas. The proposed method can achieve good localized results. It is less noisy and the veins are more complete. Figs. 10(g)~(j) are the localized results from the models based on ResNet18, ResNet50, VGG and AlexNet. It can be seen that ResNet18 gives some artifacts in the lower boundary of the arm. ResNet50 has clearer vein patterns in some area, but its processing time increases 30% since the model has more layers, which will add burdens to the embedded system. The result from the VGG model is too blurred and some small veins have been smoothed. AlexNet almost gives nothing from the RGB image. Figs. 10(k)~(D) show another two sets of results. The proposed method still can produce better result than the other methods.



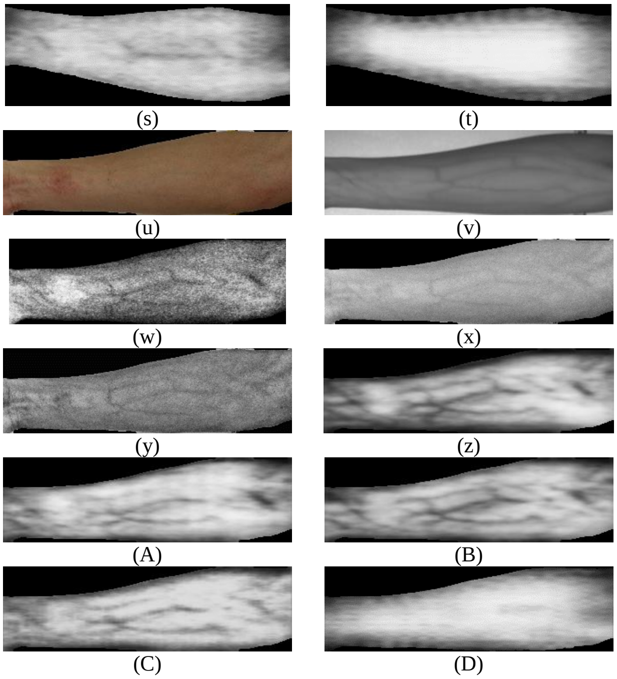


Fig. 10. Subjective evaluation on the JAI image dataset
(a) is a color skin image of an inner forearm; (b) is its corresponding NIR image; (c)~(f) are the vein localized results from the Optical method, Wiener method, Watanabe's method, and the proposed method, respectively; (g)~(j) are the vein localized results from the models based on ResNet18, ResNet50, VGG and AlexNet. (k)~(D) show another two sets of results.

Numerical measures are also adopted to evaluate the proposed vein localization model quantitatively. Only the JAI dataset has synchronous RGB/NIR images, so the objective evaluation is performed in this dataset. We use a filter bank composed of the real parts of 16 Gabor filters to the resultant images and NIR images to locate veins. Then the information maps are enhanced and binarized using Otsu's method [38]. After that veins could be obtained. An example is given in Fig. 11. Fig. 11 (a) and (b) is a pair of RGB/NIR arm images. Fig. 11 (c) is the vein images obtained from Fig. 11 (b). Figs. 11 (d)-(g) are the vein images obtained from the localized results of the optical method, Wiener method, Watanabe's method, and the proposed method; Figs. 11 (h)-(l) are those from the proposed method without perceptual loss, the proposed method based on ResNet18, ResNet50, VGG16, and AlexNet, respectively. It can be seen that with the NIR image as a benchmark, the veins obtained by the proposed method are more complete and less noisy. With the NIR images as ground truth, the precision, recall, accuracy, and F1_score are calculated from the numbers of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) vein pixels as shown in Eq.s (16)-(19) [40]. #TP represents the number of pixels correctly classified as veins whereas #FN shows that incorrectly classified as general skin. #FP indicates the number of pixels incorrectly classified as veins whereas #TN indicates that correctly classified as general skin. The fifth evaluation metric is Overlap Percentage (OP), which is defined as the intersection over union for all correct classification. As shown in Eq. (20), A_r and A_g are the areas of veins in the resultant image and ground truth image, respectively.

$$Precision = \frac{\#TP}{\#TP+\#FP} \tag{16}$$

$$Recall = \frac{\#TP}{\#TP+\#FN} \tag{17}$$

$$Accuracy = \frac{\#TP+\#TN}{\#TP+\#FP+\#TN+\#FN} \tag{18}$$

$$F1_{score} = 2 \times \frac{Precision \times Recall}{Precision+Recall} \tag{19}$$

$$OP = \frac{A_r \cap A_g}{A_r \cup A_g} \tag{20}$$

The mean values of the five measures are recorded in Table III. The best values are highlighted. It can be seen that the proposed method has the best results for most of the metrics. ResNet50 has the highest recall value, but it needs more processing time. The results of quantitative comparison show that using pixel classification methods to measure, the proposed vein localization model has promising performance.

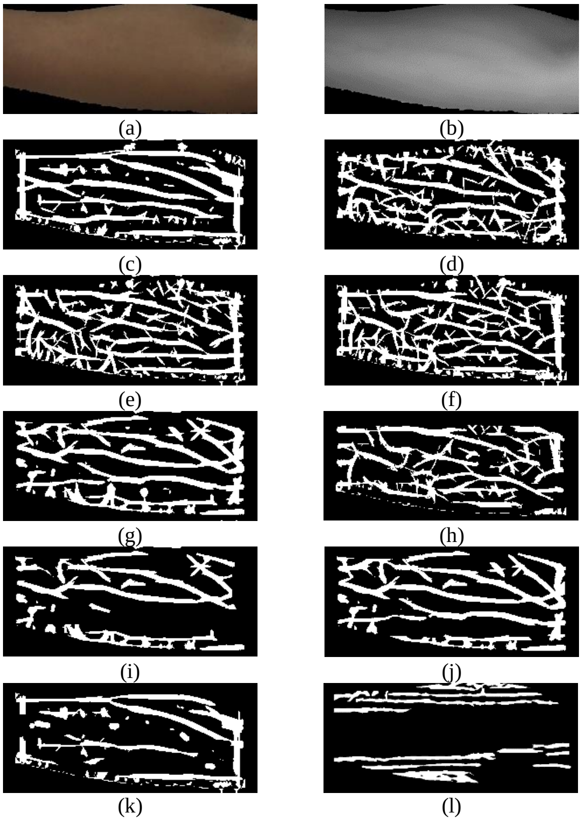


Fig. 11. Objective evaluation on the JAI image dataset
(a) and (b) is a pair of RGB/NIR arm images; (c) is the vein images obtained from (b); (d)-(g) are the vein images obtained from the localized results of the Optical method, Wiener method, Watanabe's method, and the proposed method; (h)-(l) are those from the proposed method without perceptual loss, the proposed method based on ResNet18, ResNet50, VGG16, and AlexNet, respectively.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3058014, IEEE Access

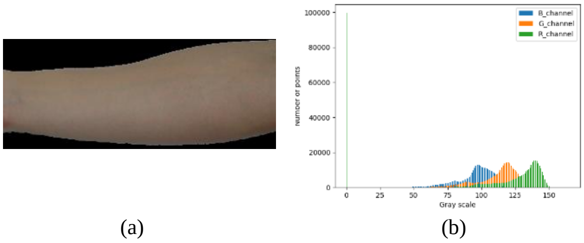
TABLE III
MEAN VALUES OF PRECISION, RECALL, ACCURACY, AND F1 SCORE

Metrics	Precision	Recall	Accuracy	F1 score	IOU
Optical method	0.6155	0.5552	0.7787	0.6075	0.4447
Wiener method	0.6038	0.5799	0.7750	0.6139	0.4497
Watanabe's method	0.5981	0.5764	0.7718	0.6033	0.4375
AlexNet	0.4072	0.4114	0.6573	0.4091	0.2580
ResNet18	0.5672	0.5701	0.7499	0.5684	0.4009
ResNet34	0.5678	0.5706	0.7502	0.5689	0.4016
ResNet50	0.5927	**0.5965**	0.7647	0.5944	0.4273
Proposed method	**0.6216**	0.5870	**0.7827**	**0.6207**	**0.4548**

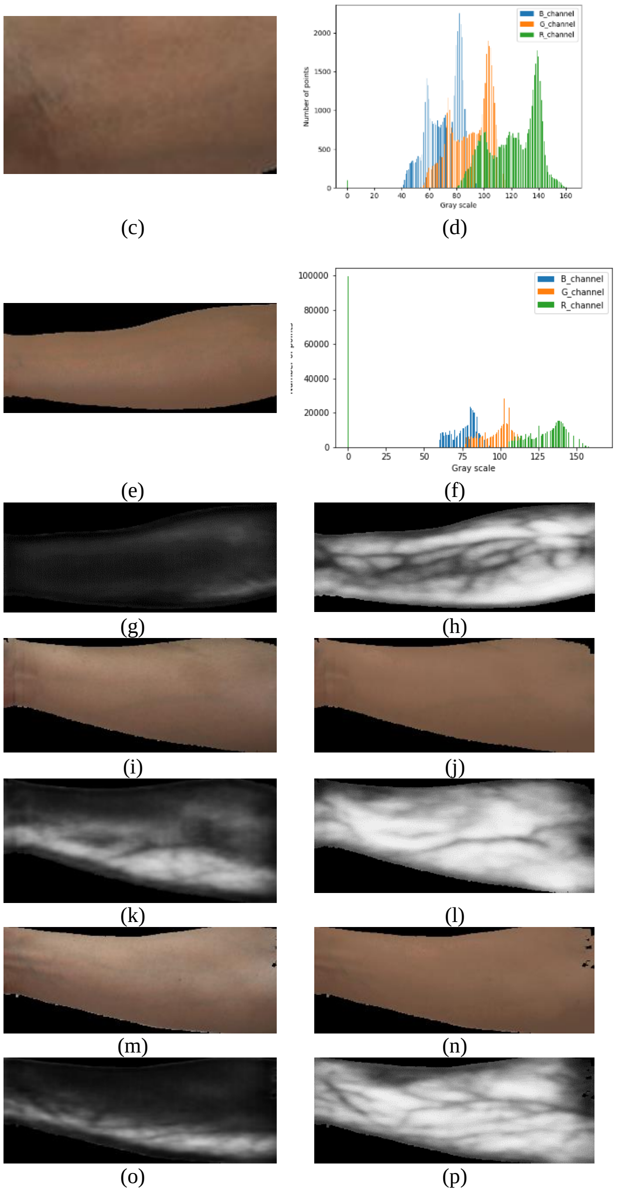
C. Evaluation of the histogram specification scheme and the perceptual loss

For the reason of different lighting conditions and camera properties, the images taken by other cameras may be very different from those taken by the JAI camera, which provides the training samples of our deep network. It may affect the performance of the model. As mentioned in Section III, we adopt a histogram specification scheme to make the model adaptive to different images. In this experiment, we use the DSLR dataset to evaluate the specification scheme.

Fig. 12(a) is the image of a right forearm taken by the DSLR camera. Fig. 12(c) is a typical skin image in the training dataset. Fig. 12(e) is the result of histogram specification on (a). Figs. 12(b), (d) and (f) are the histograms of the R, G and B channels of (a), (c) and (e), respectively. The stretching effect can be visualized from (f). The shape of the histogram is modified according to that of the target image, and the color style of the original image is transferred also. Fig. 12(g) and (h) are the localized results of (a) and (e), respectively. Obviously, the effect of (h) is much better than that in (g). Figs. 12(i)~(p) are another two sets of examples. The effect of histogram specification can be detected also.



(a) (b)



(c) (d)

(e) (f)

(g) (h)

(i) (j)

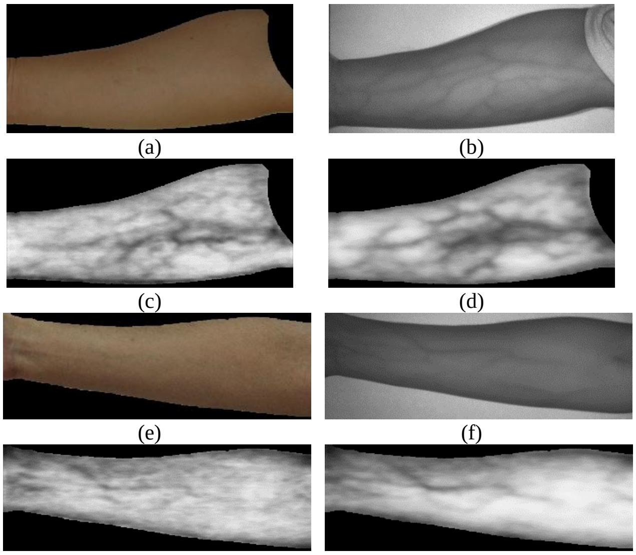
(k) (l)

(m) (n)

(o) (p)

Fig. 12. Evaluation of the histogram specification scheme
(a) is the image of a right forearm taken by the DSLR camera; (c) is a typical skin image in the training dataset; (e) is the result of histogram specification on (a); (b), (d) and (f) are the histograms of the R, G and B channels of (a), (c) and (e), respectively; (g) and (h) are the localized results of (a) and (e), respectively; (i) ~ (p) are another two sets of examples.

We also evaluated the combined loss function which includes a per-pixel loss and a perceptual loss. Fig. 13 (a) and (b) are a pair of RGB/NIR arm images; (c) is the vein image obtained from the proposed method with the per-pixel loss only; (d) is the vein image obtained from the proposed method with the combined loss. It can be seen that the localized result with the perceptual loss is less noisy and clearer than that with only the per-pixel loss.



(a) (b)

(c) (d)

(e) (f)

(g) (h)

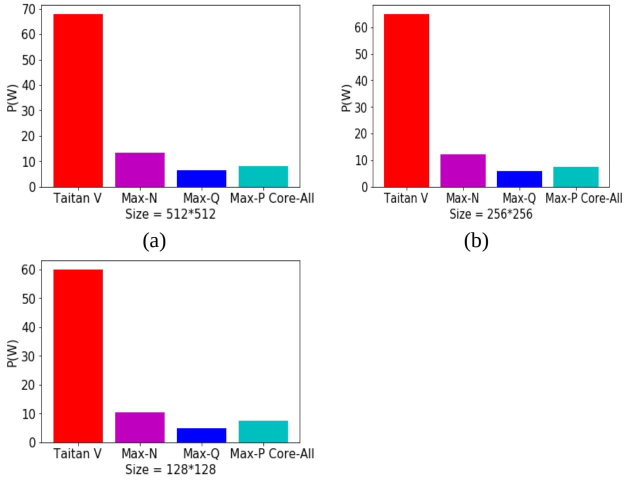
Fig. 13. Evaluation of the combined loss function
(a) and (b) are a pair of RGB/NIR arm images; (c) is the vein image obtained from the proposed method with the per-pixel loss only; (d) is the vein image obtained from the proposed method with the combined loss; (e)-(h) are another set of examples.

D. Evaluation of energy consumption

Energy consumption is one of the main limiting factors for deep learning applications on embedded system. Although CNNs offer lots of opportunities for image processing tasks, their practical use and expansion are highly dependent on the development of hardware-oriented algorithms and its implementation. Instead of the computation itself, the energy consumption of CNNs is mainly determined by data movement [28]. Luckily, the costliest operations in data movement are greatly parallel.

The power consumption of the proposed vein localization model running on the GPU server (Titan XP) and the Jetson TX2 is shown in Fig. 14, and the time consumption is summarized in Table IV. It can be seen that for the same task, Titan XP consumes more power and less time. For Jetson TX2, the consumed power and time are measured as a function of the image size for different operation modes of the development kit. The full clocks mode is activated, so all cores have to run at the maximum speed. With the presence of more veins in the image, power consumption also increases. As can be observed, the Max-Q mode limits the clocks to ensure operation in the most efficient range. It is the most cost-effective mode in terms of energy, ranging from 4.8 to 6.6 W. On the contrary, Max-N is the most expensive mode with a maximum consumption of 13.3 W since all CPUs and GPU run at maximum clock speeds. Max-P Core-all mode is a balance between both, running all CPUs.

It can be seen that the proposed vein localization model can solve the application problem in real time with low memory usage. It is implemented on a hardware platform with a small dimension, an affordable price and a low-power consumption.



(c)

Fig. 14. Power consumption of the vein localization method
(a), (b) and (c) are the power consumptions of the proposed method under different operation modes for different image sizes.

TABLE IV
TIME CONSUMPTION OF THE VEIN LOCALIZATION SYSTEM ON JETSON TX2

Mode/Metrics	NVIDIA Titan Xp	Jeston Max-N	Jeston Max-P Core-All	Jeston Max-Q
Size512*512 Time (s)	1.38	3.03	5.74	8.40
Size256*256 Time (s)	0.53	1.27	1.81	3.20
Size128*128 Time (s)	0.31	0.53	0.81	1.39

V. CONCLUSION

Venipuncture is one of the most important procedures for both medical testing and treatment. Localizing veins by computerized methods has been an emerging research area with the evolution of computer vision, especially deep learning methods. In this paper, an end-to-end solution for real time deep learning-based vein localization from color skin images in an embedded and low-power platform was presented. A deep network model consisting of a fully convolutional network, a dilated convolution module, and a transposed convolution module was proposed to extract mapping information from a synchronized RGB/ NIR image database. A combined loss function including a per-pixel loss and a perceptual loss, and a histogram specification scheme were also presented. NVIDIA Jetson TX2 development kit was chosen for the embedded system because it allows development from beginning to end, including wireless connection. It was powered by a LiPo battery and used the touch screen to operate. Other strengths of the platform are price and dimensions. Both the localization model and the embedded system were tested for different image datasets. Qualitative and quantitative results showed that the developed system has promising perspective in the real time medical scenarios. In the future work, we will use the TensorRT acceleration engine to further accelerate the model.

REFERENCES

[1] C. Huang et al., "Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China," The Lancet, vol. 395, no. 10233, pp. 497-506, Jan. 2020. DOI: 10.1016/S0140-6736(20)30183-5.

[2] COVID-19 epidemic tracking, https://www.bing.com/covid/.

[3] C. Wang et al., "A novel coronavirus outbreak of global health concern," The Lancet, vol. 395, no. 10233, pp. 470-473, Jan. 2020. DOI: 10.1016/S0140-6736(20)30185-9.

[4] "The assessment Strategy for Cannulation and Venipuncture," [Online]. Available: http://www.ruh.nhs.uk/training/prospectus/clinical_skills/documents/cannulation_and_venepuncture_workbook.doc.

[5] A. F. Jacobson, E. H. Winslow, "Variable's influencing intravenous catheter insertion difficulty and failure: an analysis of 339 intravenous catheter insertions," Heart & Lung J., vol. 34, no. 5, pp. 345-359, Oct. 2005. DOI: 10.1016/j.hrtlng.2005.04.002.

[6] D. Mbamalu, A. Banerjee, "Methods of obtaining peripheral venous access in difficult situations," Postgrad Med J., vol. 75, no. 886, pp. 459-462, 1999.

[7] M. Asrar, A. Al-Habaibeh, M. R. Houda, "A comparative study between visual, near infrared and infrared images for the detection of veins for intravenous cannulation," in Proc. AITA 2015 - Advanced Infrared Technology and Applications, Pisa, Italy, 2015.

[8] A. Shahzad, M. N. M. Saad, N. Walter, et al., "A Review on subcutaneous veins localization using imaging techniques," Current Medical Imaging Reviews, vol. 10, no. 2, pp. 125-132, 2014.

[9] Y. Yamagami, S. Ueki, K. Matoba, et al., "Effectiveness of ultrasound-guided peripheral intravenous cannulation in pediatric patients aged under three years: a systematic review protocol," JBI Database of Systematic Reviews and Implementation Reports, 2018, 16(1):35-38.

[10] K. M. Englund, M. Rayment, "Nutcracker syndrome: A proposed ultrasound protocol," Australian Journal of Ultrasound in Medicine, vol. 21, no. 2, pp.75-78, 2018.

[11] NIR vs. Ultrasound vs. Transillumination for Vein Access, https://www.veinlite.com/blog/post/nir-ultrasound-transillumination-vein-access/.

[12] N. J. Cuper, J. C. Graaff, R. M. Verdaasdonk et al., "Near-infrared Light Device Can Improve Intravenous Cannulation in Critically Ill Children," Pediatrics & Neonatology, vol. 54, no. 3, pp.194-197, Jun. 2013.

[13] C. T. Pan, M. D. Francisco, C. Yen , et al., "Vein pattern locating technology for cannulation: a review of the low-cost vein finder prototypes utilizing near Infrared (NIR) light to improve peripheral subcutaneous vein selection for phlebotomy," Sensors, vol. 16, pp. 3573-3589, 2019.

[14] M. Asrar, A. Al-Habaibeh, M. Houda. "Innovative algorithm to evaluate the capabilities of visual, near infrared, and infrared technologies for the detection of veins for intravenous cannulation," Applied Optics, vol. 55, no. 34, pp. 67-75, 2016.

[15] QU Xuemin, BIAN Zhengzhong. Development of Ultrasonic Positioning Detector for Vein. Chinese Journal of Medical Physics, 2000, 17(01):32-33. (in Chinese).

[16] S. S. Pavanjeet, N. Walter, A. Shahzad and N. M. Saad, "Optimum illuminant determination based on multispectral spectroscopy for enhanced vein detection," 2015 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), Kuala Lumpur, 2015, pp. 174-179.

[17] A. Shahzad, N. Walter, A. S. Malik, N. M. Saad and F. Meriaudeau, "Multispectral venous images analysis for optimum illumination selection," 2013 IEEE International Conference on Image Processing, Melbourne, VIC, 2013, pp. 2383-2387.

[18] S. Çolak, Ö. F. Özdemir and Y. S. Akgül, "A stereo camera system for palm vein biometrics," 2016 24th Signal Processing and Communication Application Conference (SIU), Zonguldak, 2016, pp. 1825-1828.

[19] Ö. F. Özdemir, S. Çolak and Y. S. Akgül, "Regression based stereo Palm Vein extraction and Identification system," 2016 24th Signal Processing and Communication Application Conference (SIU), Zonguldak, 2016, pp. 1529-1532.

[20] N. S. Gnee, "A study of hand vein, neck vein and arm vein extraction for authentication," 2009 7th International Conference on Information, Communications and Signal Processing (ICICS), Macau, 2009, pp. 1-4.

[21] T. Ahmed et al., "Real time injecting device with automated robust vein detection using near infrared camera and live video," 2017 IEEE Global Humanitarian Technology Conference (GHTC), San Jose, CA, 2017, pp. 1-8.

[22] C. Tang, A. W. K. Kong and N. Craft, "Uncovering vein patterns from color skin images for forensic analysis," CVPR 2011, Providence, RI, 2011, pp. 665-672. DOI: 10.1109/CVPR.2011.5995531.

[23] C. Tang, H. Zhang, A. W. Kong, "Using multiple models to uncover blood vessel patterns in color images for forensic analysis," Information Fusion, vol. 32, Part B, pp. 26–39, 2016.

[24] J. H. Song, C. Kim and Y. Yoo, "Vein Visualization Using a Smart Phone With Multispectral Wiener Estimation for Point-of-Care Applications," in IEEE Journal of Biomedical and Health Informatics, vol. 19, no. 2, pp. 773-778, March 2015.

[25] T. Watanabe and T. Tanaka, "Vein authentication using color information and image matching with high performance on natural light," 2009 ICCAS-SICE, Fukuoka, 2009, pp. 3625-3629.

[26] G. Ma, B. Wang, C. Tang, "Uncovering vein pattern using generative adversarial network," Eleventh International Conference on Digital Image Processing (ICDIP 2019): 111793R, 2019.

[27] B. Blanco-Filgueira, D. García-Lesta, M. Fernández-Sanjurjo, V. M. Brea and P. López, "Deep Learning-Based Multiple Object Visual Tracking on Embedded System for IoT and Mobile Edge Computing Applications," in IEEE Internet of Things Journal, vol. 6, no. 3, pp. 5423-5431, June 2019. DOI: 10.1109/JIOT.2019.2902141.

[28] M. Goyal, N. D. Reeves, S. Rajbhandari and M. H. Yap, "Robust Methods for Real-Time Diabetic Foot Ulcer Detection and Localization on Mobile Devices," in IEEE Journal of Biomedical and Health Informatics, vol. 23, no. 4, pp. 1730-1741, July 2019. DOI: 10.1109/JBHI.2018.2868656.

[29] T. M. Hoang, S. H. Nam and K. R. Park, "Enhanced Detection and Recognition of Road Markings Based on Adaptive Region of Interest and Deep Learning," in IEEE Access, vol. 7, pp. 109817-109832, 2019.

[30] J. M. Haut, S. Bernabé, M. E. Paoletti, R. Fernandez-Beltran, A. Plaza and J. Plaza, "Low–High-Power Consumption Architectures for Deep-Learning Models Applied to Hyperspectral Image Classification," in IEEE Geoscience and Remote Sensing Letters, vol. 16, no. 5, pp. 776-780, May 2019.

[31] Forensic Skin Image Databases of Nanyang Technological University in Singapore , http://forensics.sce.ntu.edu.sg/.

[32] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition", 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 4, pp. 770-778, 2016.

[33] F. Yu and V. Koltun, "Multi-Scale Context Aggregation by Dilated Convolutions," in Proceedings of the International Conference on Learning Representations (ICLR), 2016, pp. 1–13. Benaim, S. and Wolf, L., "One-sided unsupervised domain mapping," Proc. NIPS, 752-762 (2017).

[34] J. Johnson, A. Alahi, and F. F. Li, "Perceptual losses for real-time style transfer and super-resolution," Proc. ECCV, 694-711 (2016).

[35] S. Benaim and L. Wolf, "One-sided unsupervised domain mapping," Proc. NIPS, 752-762 (2017).

[36] W. Zhang, D. Xin, H. Xiong, W. Zhu, J. Qiu and C. Wang, "A statistical weighting average approach for cognitive radio networks," 2016 19th International Symposium on Wireless Personal Multimedia Communications (WPMC), Shenzhen, 2016, pp. 74-78.

[37] Stochastic Weight Averaging in PyTorch, https://pytorch.org/blog/stochastic-weight-averaging-in-pytorch/

[38] G. Thomas, D. Flores-Tapia and S. Pistorius, "Histogram Specification: A Fast and Flexible Method to Process Digital

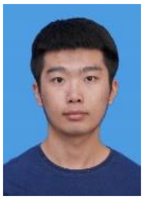
Images," in IEEE Transactions on Instrumentation and Measurement, vol. 60, no. 5, pp. 1565-1578, May 2011. DOI: 10.1109/TIM.2010.2089110.

[39] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms." IEEE Transactions on Systems, Man, and Cybernetics. vol. 9, no. 1, pp. 62–66, 1979.

[40] F1 Score. Accessed: Mar. 5, 2019. [Online]. Available: https://en.wikipedia.org/wiki/F1_score.

Chaoying Tang (S’11–M’15) received her B.Eng. and M.Eng. degrees, both in Automation, from Nanjing University of Aeronautics and Astronautics, China. She got her Ph.D. degree from the School of Computer Science and Engineering, Nanyang Technological University, Singapore in 2013. Now she is an Associate Professor with the Department of Automatic Control, College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, China. Her research interests include image processing, pattern recognition and biometrics.

Shuhang Xia received the B.E. in Automation, from Nanjing University of Aeronautics and Astronautics, in 2018. Now he is pursuing the M.E. at Nanjing University of Aeronautics and Astronautics, His research interests include image processing and pattern recognition.



Mengen Qian received the B.E. in Automation, from Yangzhou University, in 2019. Now he is pursuing the M.E. at Nanjing University of Aeronautics and Astronautics. His research interests include image processing and deep learning.

Biao Wang received B. E. in Aeroengine Control, M. E. in Aeroengine, and Ph.D. in Guidance, Navigation and Control from Nanjing University of Aeronautics and Astronautics (NUAA), respectively, in 1997, 2000, and 2004. Now he is an associate professor of NUAA. His research interests include flight control and visual guidance for unmanned aerial vehicles.