NIR-ISL 2021 Paper Form

Jalil Nourmohammadi Khiarak\*1, Farhang Jaryani1, Samaneh Salehi Nasab2, Seyed Naeim Moafinejad2, Yasin Amini2, Morteza Noshad2

\*1Warsaw University of Technology,Institute of Control and Computation Engineering, Poland

[Jalil.Nourmohammadi@elka.pw.edu.pl](mailto:abhijit.das@inria.fr)

1Arak University,Faculty of Engineering, Iran.

[S70837198@postdoct.araku.ac.ir](mailto:S70837198@postdoct.araku.ac.ir)

2School of Engineering, Lorestan University, Aleshtar Campus,Aleshtar, Iran.

[salehinasab.s@lu.ac.ir](mailto:salehinasab.s@lu.ac.ir)

2Department of Physics, Shahid Beheshti University, G.C. Evin, Tehran 1983963113, Iran  
[s.naeim.moafi.n@gmail.com](mailto:s.naeim.moafi.n@gmail.com)

2Department of Computer Engineering, Faculty of Electrical and Computer Engineering, University of Kharazmi, Iran

[amini.yasin1995@gmail.com](mailto:amini.yasin1995@gmail.com)

2Senior Research Scientist, [Stanford University](https://scholar.google.com/citations?view_op=view_org&hl=en&org=8539678734835078480), USA

noshad@stanford.edu

1. **Describe your approach in short.**

The proposed approach employs a U-Net inspired model conditioned on MobileNetV2 class features to segment iris, inner boundary, and outer boundary in an eye, where two-stage fine-tuning was applied to the MobileNetV2 model. The data was augmented by different models.

1. **Describe your approach in more detail.**

In our method, we used U-Net [1] with Pretrained MobileNetV2 [2]. The U-Net is based on the fully convolutional network and we modified its architecture to work with fewer training samples and to achieve more accurate segmentation. We used the pre-trained weights provided with MobileNetV2 for the ImageNet dataset [3] and fine-tuned it on the iris recognition and localization domain. To provide domain adaptation, we fine-tuned the MobileNetV2 model on the provided data for NIR-ISL 2021 from the CASIA-Iris-Asia, CASIA-Iris-M1, and CASIA-Iris-Africa. We also augmented the data by performing left-right flips, rotation, zoom, and brightness. MobileNetV2 has less parameters, due to which it is easy to train. For the binary masks, we chose the binarization threshold by iterating over the images in the provided dataset and applied the threshold that achieved the highest F1-score.

1. **Short name of your approach.**

Option 1: KartalOL-Net

Option 2: Mob-U-Net

1. **Did you normalize the eye images in any way? If so, what procedure did you use? Please provide a description .**

Yes. We have normalized the images to [0~1].

For iris segmentation, we have cropped images based on pupil centrum estimation and then we have trained network to segment irises.

Also, to extract inner and outer boundary, we have applied canny edge detection on images before training.

1. **Did you use RGB or grey-scale images?**

Grey-scale images.

1. **Did you use any form of data preprocessing or data augmentation? If so, what did you use?**

We performed data augmentation and created 36 images per original image in the NIR-ISL 2021 training dataset. For augmentation, we generated different versions of the image and used both in our model training. We also resized the images to 224\*224 size.

1. **What loss function or learning objectives did you use?**

We used a Dice coefficient to measure overlap of between two images as loss function. Dice coefficient term is standard in segmentation tasks and it is ranges from 0 to 1. Zero indicate worse overlap in contract one indicates perfect overlap.

1. **If you used training, how many parameters had to be learned during the training stage?**

Modified deep neural network requires training and consists of 6,504,227 total parameters and as result 4,660,323 trainable parameters.

1. **If you used a training stage, what was the size of the model weights when saved on disk (this is usually important for deep learning models)?**

For our MU-Net model, the size of the model weights was 63.7 MB.

1. **What kind of hardware did you use for the experiments (CPU, GPU, RAM, other relevant info)?**

CPU: AMD Ryzen Threadripper 1920x

RAM: Corsair CMW64GX4M4C3200C16 RGB Pro 64GB (4 x 16GB) DDR4 3200 MHz C16

GPU: NVIDIA Quadro P5000

We used the GPU for CNN model training.

1. **What programming language did you use (Matlab, Python, C, anything else? / compiled vs. interpreted)?**

For model training, the binarization threshold computation and thresholding process we used the Keras and Tensorflow framework and Python.

[1] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." *International Conference on Medical image computing and computer-assisted intervention*. Springer, Cham, 2015.

[2] Sandler, Mark, et al. "Mobilenetv2: Inverted residuals and linear bottlenecks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.

[3] Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." 2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009.