5.2.3. Mammoth (Junyan Wu et al.)

Wu et al. proposed a unified framework that combines deep feature extractor and statistical feature blending to automatically predict the DR and DME severity scores. For DME, they used DenseNet (Iandola et al., 2014) to directly predict the severity score. Whereas for DR, Kaggle training dataset was used to pretrain the DenseNet model through a dynamic sampling mechanism to balance the training instances and later fine-tuned using IDRiD dataset. Initially, the background of all images was cropped and resized to 512 × 512 pixels. Later, morphological opening and closing are utilized to preserve bright and dark regions. For instance, the morphological opening can erase the EXs and highlight the MAs. Whereas, the closing operation can remove MAs and preserve EXs. These operations can be used to denoise specific levels of classifications, for example, the risk of DME only depends on the location of the EXs. Further, several standard data augmentation methods (as shown in Table 6) are also employed. Mean Squared Error (MSE) and cross-entropy with five classes were the loss functions employed to train the network and SGD for optimization. The initial learning rate was set to 0.0005 with a decrement of 0.1 after every 30 epochs. The initial training was done by 200 epochs and fine-tuning by 50 epochs. Afterward, the last layer was removed before the final prediction, and its statistical features were aggregated together into a boosting tree. Specifically, 50 pseudo-random augmentations were performed to get 50 outputs from last second FC layer (size of 4096), then the mean and standard deviation of 50 feature vectors for each image was computed, and both vectors were then concatenated together for training in LightGBM. The output from the second last layer of fine-tuning experiments was used to train a blending model, strategy adopted from team o_O's solution of Kaggle DR challenge. Finally, for the disease grading prediction, gradient boosting tree model was built on a combined second last layer from the pre-trained network and fine-tuned network.

5.2.4. Harangim1 (Balazs Harangi et al.)

Harangi et al. proposed an approach for the classification of retinal images via the fusion of two AlexNet (Krizhevsky et al., 2012) and GoogLeNet (Szegedy et al., 2015). For this aim, they removed FC and classification layers and interconnect them by inserting a joint FC layer followed by the classic softmax/ classification layers for the final prediction. In this way, single network architecture was created which allows to train the member CNN's simultaneously. For each $I^{(n)}$, let us denote the outputs of the final FC layers of the member CNN's by $\hat{O_1}^{(n)}, \hat{O_2}^{(n)}$. The FC layer of their ensemble aggregates them via

$$\acute{O}^{(n)} = A_1\hat{O_1}^{(n)} + A_2\hat{O_2}^{(n)} \tag{6}$$

where weight matrices A_1, A_2 were of size 5 × 5 and initialized as

$$A_1 = A_2 = \begin{bmatrix} 1/5 & 0 & 0 & 0 & 0 \\ 0 & 1/5 & 0 & 0 & 0 \\ 0 & 0 & 1/5 & 0 & 0 \\ 0 & 0 & 0 & 1/5 & 0 \\ 0 & 0 & 0 & 0 & 1/5 \end{bmatrix} \tag{7}$$

The last two layers of the ensemble were a Softmax and a classification one. Let $O_{SM}^{(n)}$ be an output of a former layer, the MSE was used for optimization as a loss function:

$$MSE = \frac{1}{2N}\Sigma_{n=1}^{N}(\acute{O}_{SM}^{(n)} - O^{(n)})^2 \tag{8}$$

During the training phase, back-propagation is applied to minimize the loss via adjusting all parameters of member CNNs and weight matrices A_1, A_2.

For the grading of DME, the final layers of member CNNs consist of 3 neurons, and weight matrices A_1, A_2 were 3 × 3, initialized as

$$A_1 = A_2 = \begin{bmatrix} 1/3 & 0 & 0 \\ 0 & 1/3 & 0 \\ 0 & 0 & 1/3 \end{bmatrix} \tag{9}$$

For training, they merged IDRiD and Kaggle training set. The parameters of architectures were found by SGD algorithm in 189 and 50 epochs respectively for DR and DME classification tasks. Learning rate was set to 0.0001. Training times required on the datasets for DR and DME were 96.6 (189 epochs) and 23.4 (50 epochs) hours respectively. Implementation of this work was done in MATLAB 2017b. The training was performed using an NVIDIA TITAN X GPU card with 7 TFlops of single-precision performance, 336.5 GB/s of memory bandwidth, 3072 CUDA cores, and 12 GB memory.

5.2.5. AVSASVA (Varghese Alex et al.)

Alex et al. used ensembles of pre-trained CNNs (on ImageNet dataset), namely, ResNets (He et al., 2016) and DenseNets (Iandola et al., 2014) for the task of disease grading. For DR grading, two ensembles of CNN's namely "primary" and "expert" classifiers were used. The primary classifier was trained to classify a fundus image as one of the 4 classes viz; Normal, Mild NPDR, Moderate NPDR or S-(N)-PDR, a class formed by clubbing Severe NPDR and PDR. The expert classifier was trained exclusively on Severe NPDR or PDR images and was utilized to demarcate the input image as one of the aforementioned classes. During inference, each fundus image was resized to a dimension of 256 × 256 pixels. For the task of grading of DR in fundus images, they used test time augmentation through the "Ten Crop" function defined in PyTorch. The images were first passed through the primary classifier and then through the expert classifier, only if the image was classified as S-(N)-PDR by the primary classifier. The final prediction was achieved by using a majority voting scheme.

For DME grading, two ensembles were trained in a one versus rest approach. Ensemble 1 was trained to classify the input as either "image with no apparent EXs" (Grade 0) or "presence of EXs in image" (Grade 1 & Grade 2), while the Ensemble 2 was trained to classify an image as "Grade 2" DME or not (Grade 0 & Grade 1). During inference, the resized images were fed to both ensembles, and the final prediction was obtained by combining the two predictions by utilizing a set of user-defined rules. Briefly, the user-defined rules were: an image was classified as Grade 0 DME if ensemble 1 and ensemble 2 predict the absence of EXs and the absence of grade 2 DME respectively. A scenario wherein ensemble 2 predicts the presence of grade 2 DME, images were classified under category "Grade 2 DME" irrespective of the prediction from ensemble 1. Lastly, images were classified as Grade 1 DME if none of the above conditions were satisfied.

Both models for DR and DME were initialized with the pretrained weights and the parameters of networks were optimized by reducing cross-entropy loss with ADAM as an optimizer. The learning rate was initialized to 10^{-3} for DR and 10^{-4} for DME. For DR, the learning rate was reduced by a factor of 10% every instance when the validation loss failed to drop. Each network was trained for 30 epochs and the model parameters that yielded the lowest validation loss were used for inference. For DME, the learning rate was annealed step-wise with a step size of 10 and the multiplicative factor of learning rate decay value of 0.9.

5.2.6. HarangiM2 (Balazs Harangi et al.)

Harangi et al. combined self-extracted, CNN-based features with traditional, handcrafted ones for disease classification. They modified AlexNet (Krizhevsky et al., 2012) to allow the embedding of handcrafted features via the FC layer. In this way, they created a network architecture that could be trained in the usual way and additionally uses domain knowledge. They extended the FC layer, to get FC_{fuse}, originally containing 4096 neurons of AlexNet by adding 68-dimensional vector containing handcrafted features. Then, the 4164×5 (or 4164×3 for DME) layer FC_{class} was considered for DR (or DME) classification task. In this way, both final weighings FC_{class} of handcrafted features were obtained and the 4096 AlexNet features were trained by backpropagation.

To obtain 68 handcrafted features used by CNN, they employed one image level and two lesion-specific methods. The amplitude-frequency modulation (AM-FM) method extracts information from an image by decomposing its green channel at different scales into AM-FM components (Havlicek, 1996). As a result, a 30-element feature vector was obtained, which reflects the intensity, geometry, and texture of structures contained in the image (Agurto et al., 2010). Whereas to extract features related to the lesions MA and EX, they employed two detector ensembles (Antal and Hajdu, 2012; Nagy et al., 2011), which consist of a set of $<$preprocessing method (PP), candidate extractor (CE)$>$ pairs organized into a voting system. Such a $<$PP, CE$>$ pair was formed by applying PP to the retinal image and CE to its output. This way, a $<$PP, CE$>$ pair extracts a set of lesion candidates from the input image, acting as a single detector algorithm. They used the output of these ensembles to obtain 38 features related to the number and size of MA's and EX's. Parameters of the architectures were optimized by SGD algorithm in 85 and 50 epochs for DR and DME respectively. Training times were 83.1 (85 epochs) and 46.2 (50 epochs) hours on the datasets for DR and DME. Implementation of this work was done in MATLAB 2017b. Training has been performed using an NVIDIA TITAN X GPU card with 7 TFlops of single-precision, 336.5 GB/s of memory bandwidth, 3072 CUDA cores, and 12 GB memory.

5.3. Sub-challenge – 3: Optic disc and fovea detection

For a given image, this task seeks to get a solution to localize the OD and Fovea. Further, it seeks to get the probability of a pixel being OD (OD segmentation). Summary of approaches is detailed as follows:

5.3.1. Deepdr (Ling Dai et al.)

Dai et al. proposed a novel deep localization method, which allows coarse-to-fine feature encoding strategy for capturing the global and local structures in fundus images, to simultaneously model two-task learning problem of the OD and fovea localization. They took advantage of prior knowledge such as the number of landmarks and their geometric relationship to reliably detect the OD and fovea. Specifically, they first designed a global CNN encoder (with a backbone network of ResNet-50 (He et al., 2016)) to localize the OD and fovea centers as a whole by solving a regression task. All max-pooling layers were replaced with average pooling layers as compared to original ResNet architecture, due to the fact that max-pooling could lose some useful pixel-level information for regression to predict the coordinates. This step was used to simultaneously perform the two detection tasks, because of the geometric relationship between OD and fovea, the performance of multi-task learning is better than a single task. The predicted output coordinates of this global CNN encoder component were used for detecting the bounding boxes of the target OD and fovea. Then the current center coordinates are refined through a local encoder (with a backbone network of VGG-16 (Simonyan and Zisserman, 2014)) which only localizes the OD center or fovea center of their related bounding boxes. During the training stage, they designed an effective data augmentation scheme to solve the problem of insufficient training data. In particular, to build the training set of a local encoder, bounding boxes were randomly selected based on the ground truth, for each object several bounding boxes of different positions and scales were cropped. The local encoder can be reused multiple times to approximate the target coordinates. The local encoder was iterated twice for refining centers comprehensively. All three models were initialized from the pre-trained ImageNet network and replaced the network's last FC layer and Softmax layer by the center coordinates regressor. The regression loss for the central location was the Euclidean loss. The modified loss function for global and local encoders was $0.045(L_{OD} + L_{fovea})$ and $0.045(L_{OD}/L_{fovea})$ respectively. Where L_{OD} and L_{fovea} are losses for OD and fovea, and scaling factor was introduced since the original Euclidean distance is too large in practice to converge. The proposed learning model was implemented in Caffe framework and trained using SGD with momentum. The FC layers for center regression were initialized from zero-mean Gaussian distributions with standard deviations 0.01 and 0.001. Biases were initialized to 0. The global encoder was trained for 200 epochs, local encoders (OD and fovea both) for 30 epochs respectively. The batch size for global encoder was 16, and 64 for the other two local encoders. The learning rate was set as 0.01 and was divided by 10 when the error plateaus.

5.3.2. VRT (Jaemin Son et al.)

Son et al. proposed an OD segmentation model consisting of U-Net (Ronneberger et al., 2015) and CNN that takes a vessel image and outputs 20×20 activation map whose penultimate layer is concatenated to the bottleneck layer of U-Net. Initially, original images were cropped (3500×2848 pixels), padded (3500×3500 pixels) and then resized (640×640 pixels). Each image was standardized with its mean and standard deviation (SD). When calculating the mean and SD, values less than 10 (usual artifacts in the black background) are ignored. Vessel images were prepared with an external network (Son et al., 2017). Pixel values in a vessel image range from 0 to 1. It uses external datasets DRIONS-DB (Carmona et al., 2008) and DRIVE (van Ginneken et al., 2004) available with OD and vessel ground truths respectively. For augmentation, the fundus images were affine-transformed and additionally OD was cropped and randomly placed on the image for a random number of times (0 to 5). This augmentation was done to prevent the network from segmenting OD solely by brightness. Pairs of a fundus image and vessel segmentation were provided as input and OD segmentations in the resolution of 640×640 and 20×20 pixels are given as the ground truth. Binary cross-entropy is used as a loss function for both U-Net and vessel network with the loss of $L_{total} = L_{U-Net} + 0.1 * L_{vessel}$. Total 800 epochs were trained via Adam optimizer and decreasing learning rate with hyper-parameters of $\beta_1 = 0.5, \beta_2 = 0.999$. The learning rate was $2e^{-4}$ until 400 epochs and $2e^{-5}$ until the end. Weights and biases were initialized with Glorot initialization method (Glorot and Bengio, 2010).

They also proposed a four branch model in which two branches were dedicated to the prediction of locations for OD and fovea from vessels (vessel branches) and the other two branches aim to predict the locations from both fundus and vessels (main branches). Similar to OD segmentation, penultimate layers of vessel branches were depth-concatenated to the main branches. After deriving an activation map that represents the probability of containing an anatomical landmark, a hard-coded matrix was multiplied to yield co-ordinates. Original images were cropped as in the segmentation task and standardized with an identical method and later augmented by flip and rotation to ease implementation efforts. Mean absolute error was used as loss function for both outputs with the loss of $L_{total} = L_{main} + 0.3 * L_{vessel}$. SGD was used with Nesterov momentum of 0.9 as an optimizer. Learning rate was set to 10^{-3} from 1^{st} to 500^{th} epochs and 10^{-4} from 501^{th} to 1000^{th} epochs. All implementation was done in Keras 2.0.8

with TensorFlow backend 1.4.0 using a server with 8 TITAN X (pascal). Source code is available at https://bitbucket.org/woalsdnd/isbi_2018_fundus_challenge.

5.3.3. ZJU-BII-SGEX (Xingzheng Lyu et al.)

Lyu et al. utilized Mask R-CNN (He et al., 2017) to localize and segment OD and fovea simultaneously. It scans the image and generates region proposals by 2D bounding boxes. Then the proposals were classified into different classes and compute a binary mask for each object. They firstly preprocessed the original retinal image into fixed dimensions as network input. A feature extractor (ResNet-50) with feature pyramid networks (FPN) generates feature maps at different scales, which could be used for regions of interest (ROI) extraction. Then a region proposal network (RPN) scans over the feature maps and locates regions that contain objects. Finally, a ROI head network (RHN) is employed to obtain the label, mask, and refined bounding box for each ROI. They also incorporated prior knowledge of retinal image as a post-processing step to improve the model performance. They used IDRiD dataset and two subsets in RIGA dataset (Almazroa et al., 2018) (Messidor and BinRushed, 605 images) with OD mask provided. They applied the transfer learning technique to train the model. They firstly trained the RHN network by freezing all the layers of FPN and RPN networks and then fine-tuned all layers. The model was implemented in TensorFlow 1.3 and Python 3.4 (source code was modified from Abdulla (2017)). The learning rate started from 0.001 and a momentum of 0.9 was used. The network was trained on one GPU (Tesla K80) with 20 epochs.

5.3.4. IITkgpKliv (Oindrila Saha et al.)

Saha et al. used SegNet (Badrinarayanan et al., 2015) for segmentation of lesions and OD. OD was added as an additional class in the same problem as lesion segmentation so that the model could better differentiate EXs and OD which have similar brightness levels. However, in contrast to original SegNet, the final decoder output is fed to a sigmoid layer to produce class probabilities for each pixel independently in 7 channels. Each channel has the same size as input image: 536×356 pixels and consists of activations in the range [0, 1] where 0 corresponds to background and 1 to the presence of a corresponding class. Apart from 5 classes i.e. MA, HE, SE, EX and OD, two additional classes: (i) retinal disk excluding the lesions and OD, and (ii) black background form the 7 channels. Images were downsampled to 536×356 pixels, preserving the aspect ratio. Additionally, Drishti-GS (Sivaswamy et al., 2014) dataset was used for data augmentation to account for the case of absence of lesions. Further, horizontal, vertical and 180° flipped versions of the original images were taken. The network was trained using binary cross-entropy loss function and Adam optimizer with learning rate 10^{-3} and $\beta = 0.9$. Early stopping of the training based on the validation loss is adopted to prevent overfitting. It was observed that the validation loss started to increase after 200 epochs. One more softmax layer is introduced after the Sigmoid layer for normalizing the value of a pixel for each class across channels. The segmented output is finally upsampled for each class to 4288×2848 pixels. All implementations were done in PyTorch using 2x Intel Xeon E5 2620 v3 processor with GTX TITAN X GPU 12 GB RAM and 64 GB System RAM.

5.3.5. SDNU (Xiaodan Sui et al.)

Sui et al. used Mask R-CNN (He et al., 2017) for solving all tasks in this sub-challenge. Mask R-CNN could realize accurate target detection based on proposed candidate object bounding boxes of RPN to achieve the objective of OD and Fovea localization. At the same time, it could also get the OD segment at the mask predicting branch. The head architecture of Mask R-CNN (ResNet-101 as a backbone) consists of three parallel branches for classification, bounding-box regression, and predicting mask. By this method, the localization of OD and fovea, and segmentation of OD could be achieved directly. They retrained the network to get the new weight parameter of the framework. During the training phase, the dataset of this challenge was augmented by flipping, resizing and trained by 10-fold cross-validation. After training 2000 epochs, the last trained model is obtained. They implemented this algorithm in TensorFlow and it is processed on 8 NVIDIA TITAN Xp GPUs. The experiment environment is built under Ubuntu 16.06.

5.3.6. CBER (Ana Mendonça et al.)

Mendonça et al. proposed hand-crafted features based approach for the localization and segmentation tasks in this sub-challenge. Distinct methodologies have been developed for detecting and segmenting these structures, mainly based on color and vascular information. The methodology proposed in the context of this challenge includes three inter-dependent modules. Each module performs a single task: OD localization, OD segmentation or fovea localization. While the modules responsible for the OD localization and segmentation were an improved version of two methods previously published (Mendonça et al., 2013; Dashtbozorg et al., 2015), the method proposed for fovea localization was completely new. Initially, the module associated with the OD localization receives a fundus image and segments the retinal vasculature. Afterward, the entropy of the vessel directions is computed and combined with the image intensities in order to find the OD center coordinates. For OD segmentation, the module responsible for this task uses the position of the OD center for defining the region where the sliding band filter (Pereira et al., 2007; Esteves et al., 2012) is applied. The positions of the support points which give rise to the maximum filter response were found and used for delineating the OD boundary. Since a relation between the fovea-OD distance and the OD diameter was known (Jonas et al., 2015), the module responsible for the fovea localization begins by defining a search region from the OD position and diameter. The fovea center is then assigned to the darkest point inside that region.

6. Evaluation measures

The performance of each sub-challenge was evaluated based on different evaluation metrics. Following evaluation measures were used for different sub-challenges:

6.1. Sub-challenge – 1

In this sub-challenge, the performance of algorithms for lesion segmentation tasks was evaluated using submitted grayscale images and available binary masks. As in the lesion segmentation task(s) background overwhelms foreground, a highly imbalanced scenario, the performance of this task was measured using area under precision (*a.k.a.* Positive Predictive Value (PPV)) recall (*a.k.a.* Sensitivity (SN)) curve (AUPR) (Saito and Rehmsmeier, 2015).

$$SN = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{10}$$

$$PPV = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{11}$$

The curve was obtained by thresholding the results at 33 equally spaced instances i.e. [0, 8, 16, $\cdots$, 256] in gray levels or [0, 0.03125, 0.0625$\cdots$, 1] in probabilities. The AUPR provides a single-figure measure (*a.k.a.* mean average precision (mAP)), computed over the Set-B, was used to rank the participating methods. This performance metric was used for object detection in The PASCAL Visual Object Classes (VOC) Challenge (Everingham et al., 2010).

The AUPR measure is more realistic (Boyd et al., 2013; Saito and Rehmsmeier, 2015) for the lesion segmentation performance over the area under Receiver Operating Characteristic (ROC) curve.

6.2. Sub-challenge – 2

Let the expert labels for DR and DME be represented by $DR_G(n)$ and $DME_G(n)$. Whereas $DR_O(n)$ and $DME_O(n)$ are the predicted results, then correct instance is the case when the expert label for DR and DME matches with the predicted outcomes for both DR and DME. This was done since, even with the presence of some exudation that may be categorized as mild DR, its location on the retina is also an important governing factor (to check DME) to decide the overall grade of disease. For instance, EXs presence in the macular region can affect the vision of patient to a greater extent and hence, it should be dealt with priority for referral (that may otherwise be missed or cause a delay in treatment with the present convention of only DR grading) in the automated screening systems. Hence, disease grading performance accuracy for this sub-challenge, from the results submitted in CSV format for test images (i.e. $N = 103$), is obtained by algorithm 1 as follows:

Algorithm 1: Computation of disease grading accuracy.

```
Data: Method Results and Labels with DR and DME Grading
Result: Average disease grading accuracy for DR and DME
for $n = 1, 2, \cdots, N$ do
    Correct = 0;
    if $(DR_O(n) == DR_G(n))$ and $(DME_O(n) == DME_G(n))$ then
        Correct = Correct + 1;
    end
end
Average Accuracy = $\frac{\text{Correct}}{N}$
```

6.3. Sub-challenge – 3

For the given retinal image, the objective of sub-challenge – 3 (task - 6 and 7) was to predict the OD and fovea center coordinates. The performance of results submitted in CSV format was evaluated by computing the Euclidean distance (in pixels) between manual (ground truth) and automatically predicted center location. Lower Euclidean distance indicates better localization. After determining these distances for each image in the Set-B, i.e. for 103 images, the average distance representing the whole dataset was computed and used to rank the participating methods.

The optic disc segmentation (task - 8) performance is evaluated using Jaccard index (J) (Jaccard, 1908). It represents the proportion of overlapping area between the segmented OD (O) and the ground truth (G).

$$J = \frac{|O \cap G|}{|O \cup G|} \tag{12}$$

Higher J indicates better segmentation. For the segmented results, images in range [0, 255], it was computed at 10 different equally spaced thresholds [0, 0.1, $\cdots$, 0.9] and averaged to obtain final score.

7. Results

This section reports and discusses the results of all sub-challenges. Performance of all competing solutions on the Set-B for all eight subtasks are divided into three sub-challenge categories and discussed including their leaderboard rank.

7.1. Sub-challenge – 1

In this section, we present the performance of all competing solutions for the lesion segmentation task. All results received from the participating teams were analyzed using the validation measure given in Section 6.1. This measure generated a set of precision-recall curves for each of the different techniques. Out of the total 37 teams that participated in the challenge, 22 teams participated (a complete list is available on the challenge website) in the sub-challenge–1 whose results were evaluated and ranked using the AUPR values. Amongst them, 7 teams (see Table 5) having performance within top 4 positions in either of lesion segmentation task were invited for the challenge workshop and 3 teams having overall better performance, i.e. solutions developed by the teams that ranked amongst the top three for at least three different lesion segmentation tasks, presented their work at ISBI.

Table 8 summarizes the individual performance (Off-site evaluation) of each solution listed in order of their final placement for each subtask. It also contains various approaches followed and external dataset (if any) used for training the models. A higher rank indicates more favorable performance for the individual task(s). The top-3 entries according to the individual lesion segmentation task are VRT, iFLYTEK-MIG and PATech. Some sample lesion segmentation results illustrated in Fig. 6 and their corresponding overall evaluation score from Table 8 gives a better idea of how the evaluation scores correlate with the quality of segmentation. Fig. 7 summarizes the performance of top-4 teams per lesion segmentation task. The different curves represent the performance of the participating methods for various lesions (MAs, HEs, SEs and EXs). Team VRT achieved highest AUPR score for HE and SE segmentation task. Whereas, team PATech and iFLYTEK-MIG obtained best score for EX and MA segmentation task respectively.

7.2. Sub-challenge – 2

This section presents the results achieved (On-site evaluation) by participating teams for the DR and DME grading task. It is important to note that this task was evaluated for simultaneous grading of DR and DME using the validation algorithm outlined in Section 6.2 on the Set-B. This algorithm produced an average grading accuracy of joint DR and DME on all images. Table 9 summarizes the result of teams for the on-site challenge along with the approach followed and external dataset used for training the model by respective team.

The top-performing solution at the “on-site” challenge was proposed by team LzyUNCC followed by team VRT and team Mammoth. Fig. 8 shows the average accuracy of competing solutions for the individual as well as simultaneous grading of DR and DME. Teams are observed to perform poorly in the DR grading task that reduced the overall accuracy for simultaneous grading of DR and DME. Major reason seems to be the difficult test set, difficulty in accurately discriminating the DR severity grades.

7.3. Sub-challenge – 3

This section presents an evaluation of “On-site” results for participating teams in the sub-challenge – 3, for all three subtasks. The results for subtasks of OD and Fovea center localization were evaluated by computing Euclidean distance, whereas OD segmentation results were evaluated and ranked using Jaccard similarity score as outlined in Section 6.3. Results from the on-site evaluations are reported in Table 10 and Table 11 that summarises the performance of all participating algorithms for all three subtasks.

The winning methods for localization tasks were developed by team DeepDR and team VRT, with DeepDR performing best in both