Deep Learning CAD Project Defense

Abdelrahman HABIB

Outline

1. Augmentation & Pre-processing
    a. General augmentation/pre-processing
    b. Hair augmentation (segmentation + augmentation)
2. Segmentation Masks
3. Architectures
    a. Base Architecture with visual attention blocks
    b. Ensemble 5-folds Architecture
4. Configurations (Experiments Controllers)
5. Results & Discussion
6. Grad-CAM Visualization
7. Conclusion

[1/7] Augmentation & Pre-processing (1/3)

Train

1. CropFrame [a]
2. AdvancedHairAugmentation[a]
3. HorizontalFlip
4. VerticalFlip
5. ColorJitter
6. Rotate
7. RandomBrightnessContrast
8. OneOf (MotionBlur, MedianBlur, GaussianBlur, GaussNoise)
9. CLAHE
10. Transpose
11. HueSaturationValue
12. Resize
13. Normalize

Validation | **Test**

1. CropFrame[a]
2. Resize
3. Normalize

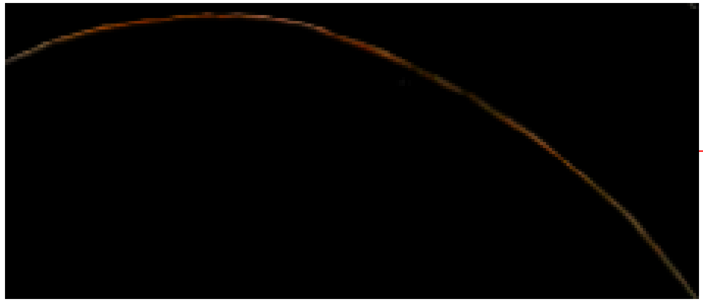
Returns:
- image
- segmentation mask [b]
- label [c]

(a) Custom pre-processing function added to the augmentation transformer with probability (p), (1. p = 100%, 2. p=50%).
(b) Depends on the experiment, we return the mask or None.
(c) Random labels for test as we don’t use it (labels are generally based on the filename (class) and index.

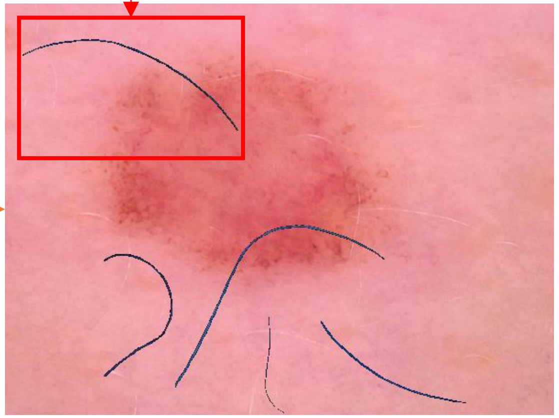
[1/7] Augmentation & Pre-processing – *Hair Augmentation* (2/3)

Sample to segment → Manual Hair Segmentation → Resized, Adjusted Contrast

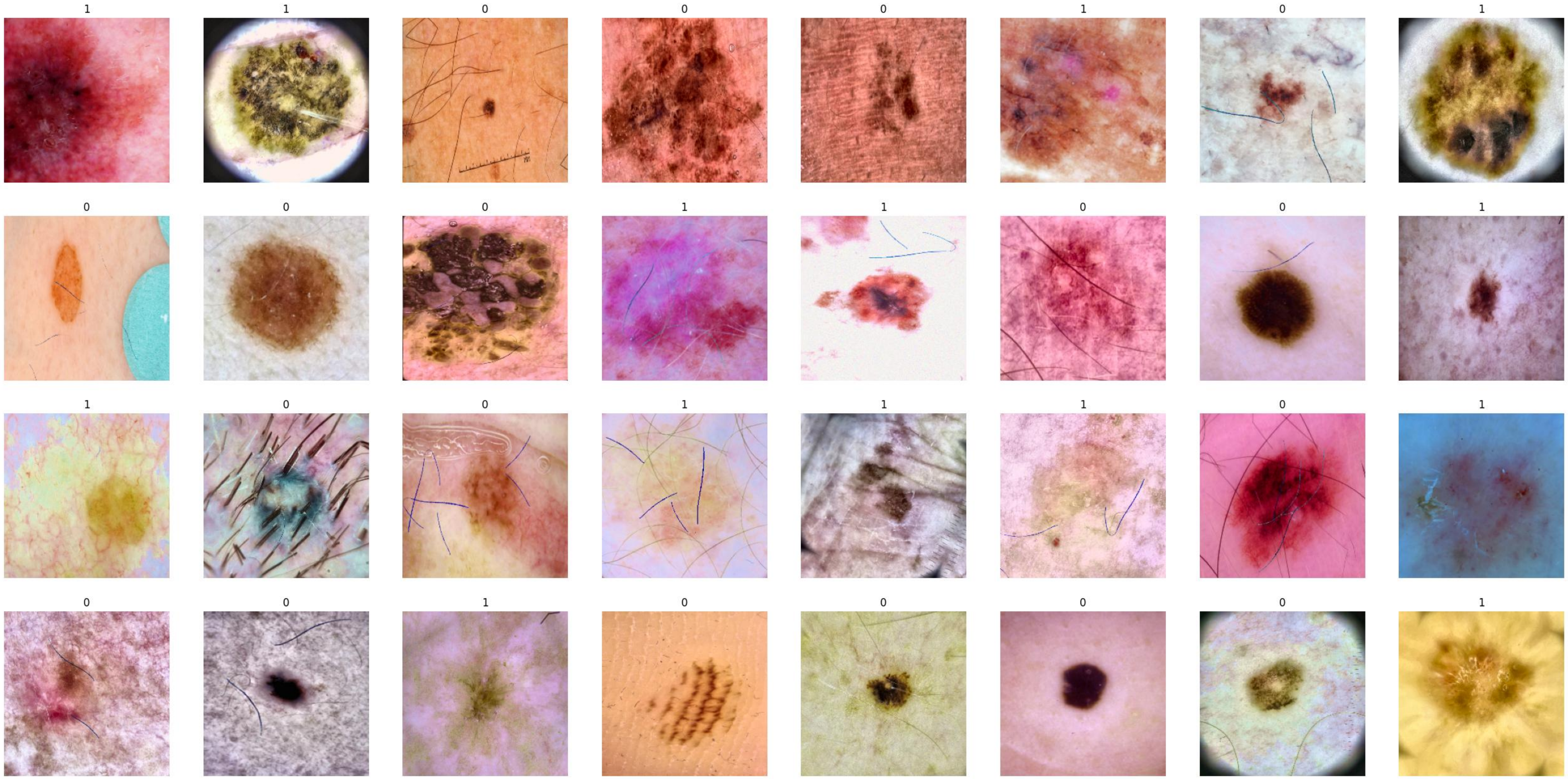




Segmented 5 hair samples from different lesions. The resize and contrast/intensity adjustment is to ensure that the augmented hair is small & dark. Hair is augmented with random rotation, flip, sizes, and maximum 5 new hairs in each image [0 to 5].

[1/7] Augmentation & Pre-processing *(Results)* (3/3)



[2/7] Skin Lesion Segmentation (1/1) (*)

Clear & simple to segment	Very small & similar objects surrounding	Multiple objects with lesion	A lot of hair on the lesion	Black corners surrounding

** Purely image processing! Detailed methodology can be found in the appendix slides, as this is an optional step in the project (yet improved the results).*

[3/7] Architectures (Base/Single Model) (1/2)

Uses **visual attention blocks**, **concatenates its output features to base model feature vector, then into classifier layer.**

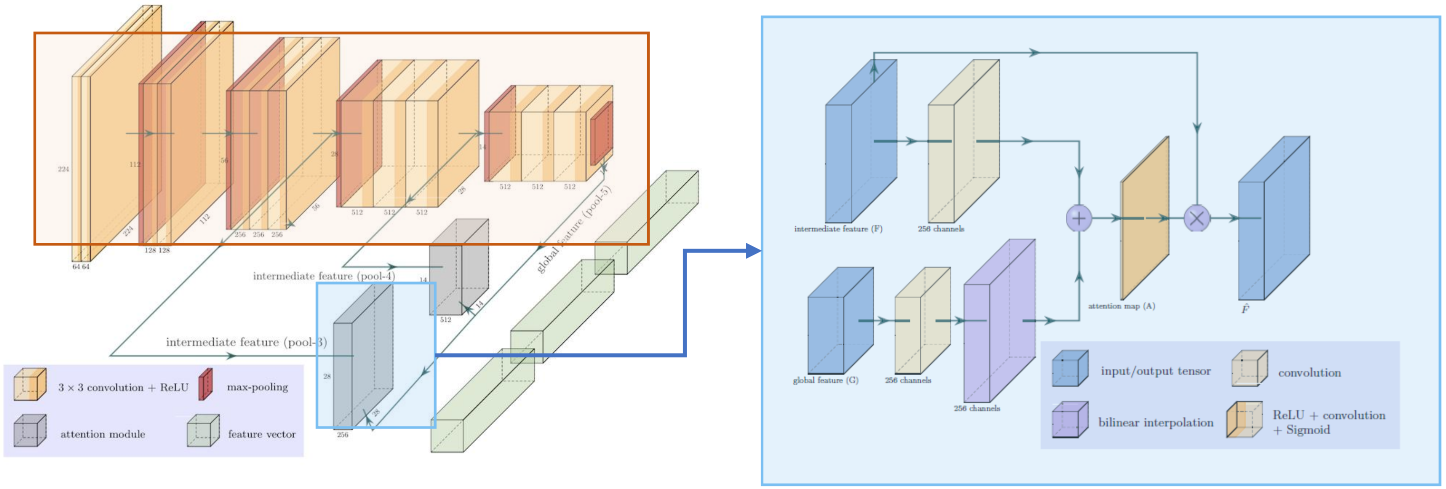
Model is trained from scratch! Loaded pre-trained weights and initialized the classifier and both attention blocks using ‘kaiming normal’.

Same base architecture for binary and multi-class problem.

Outputs: features (that we combine) + attention masks (that we use with our segmentation in the loss term)

VGG16_BN Model

Attention Blocks X 2



The authors used a single model and claimed to be the state-of-the-art with a single model, reaching auc = 88% with masks. This model can work with/without masks. Can we do better?

Yan, Y., Kawahara, J., & Hamarneh, G. (2019). Melanoma recognition via visual attention. In Information Processing in Medical Imaging: 26th International Conference, IPMI 2019, Hong Kong, China, June 2–7, 2019, Proceedings 26 (pp. 793-804). Springer International Publishing.

[3/7] Architectures (Ensemble) (2/2)

Let’s call the entire model as VGG16_BN_Attention.

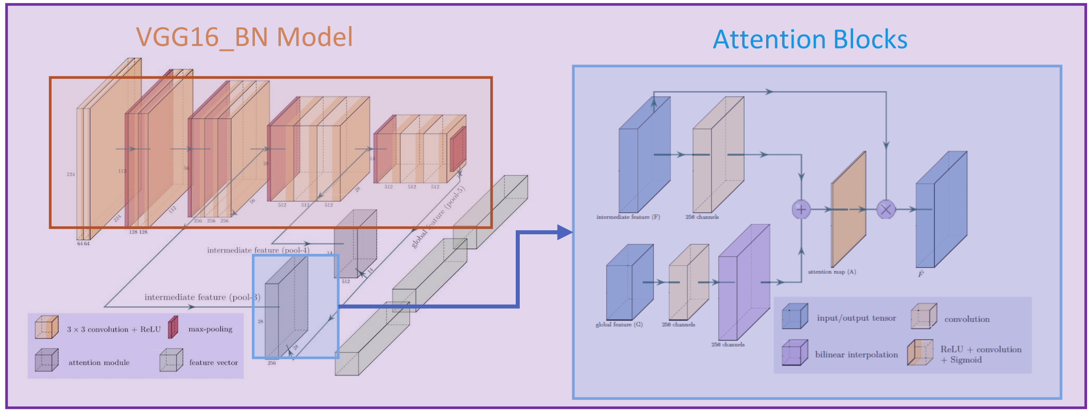
Our approach is to combine both train and valid splits, the split into k-folds using **StratifiedKFold**; k=5.

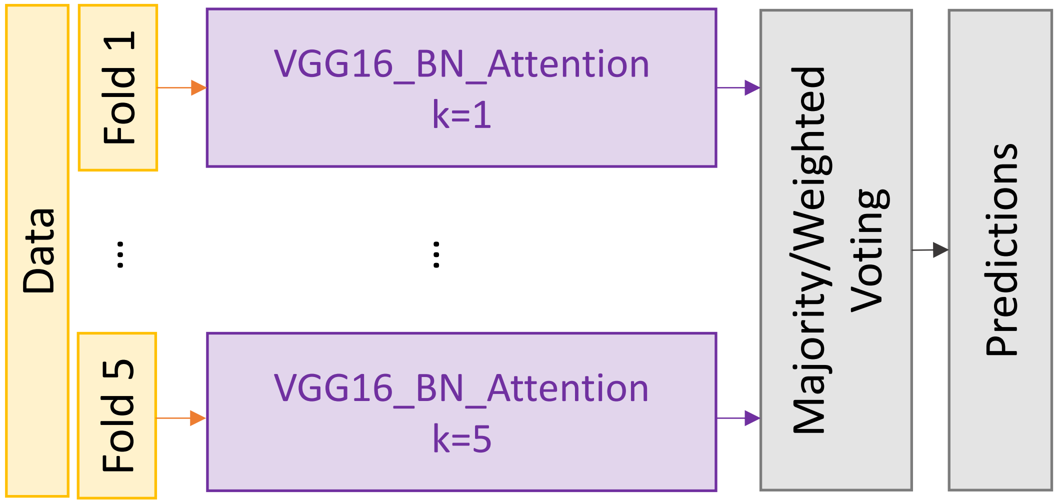
Train k-Models, on every split. **This is one way to handle imbalance of challenge 2!** Combine using majority voting.

More robust results, as the valid split is always balanced, more data to train.

Can **work with/without masks**; will be discussed later in the configuration slides.

VGG16_BN_Attention





[4/7] Configurations

	Challenge 1	Challenge 2
Epochs	50 (EarlyStopping stops with patience = 5)	
Learning Rates	0.0001 & 0.00001 (best results – avoided overfitting)	
Experiments Controllers	1. ClassifierExperiment.py (Base 1 Model / No Masks Used) 2. ClassifierExperiment**CV**.py (Base ensemble Models / No Mask Used) 3. Classifier**Seg**Experiment.py (Base 1 Model / Masks Used) 4. Classifier**Seg**Experiment**CV**.py (Base ensemble Models / Masks Used)	
Loss (No Mask, **Experiments 1, 2**)	L_{WCE} *(Weighted Cross Entropy)*	$L_{FL} = FL(p_t) = -\alpha(1-p_t)^{\gamma}\log(p_t)$* (Multi-class focal loss)
Loss (w/ Mask, **Experiments 3, 4**)	**Challenge 1:** $L = L_{WCE} + \lambda_1 Dice(SegMask, AttMask1) + \lambda_2 Dice(SegMask, AttMask2)$ **Challenge 2:** $L = L_{FL} + \lambda_1 Dice(SegMask, AttMask1) + \lambda_2 Dice(SegMask, AttMask2)$	$\lambda_1 = 0.001$ $\lambda_2 = 0.01$ $AttMask = Attention\ Mask$ $SegMask = Segmentation\ Mask$

* ***Multi-class focal loss*** *is initialized with weighted CE, making it* ***handle class imbalance for challenge 2****.*
* *The only difference between attention mask 1 and 2 are the sizes (different scale factor used).*

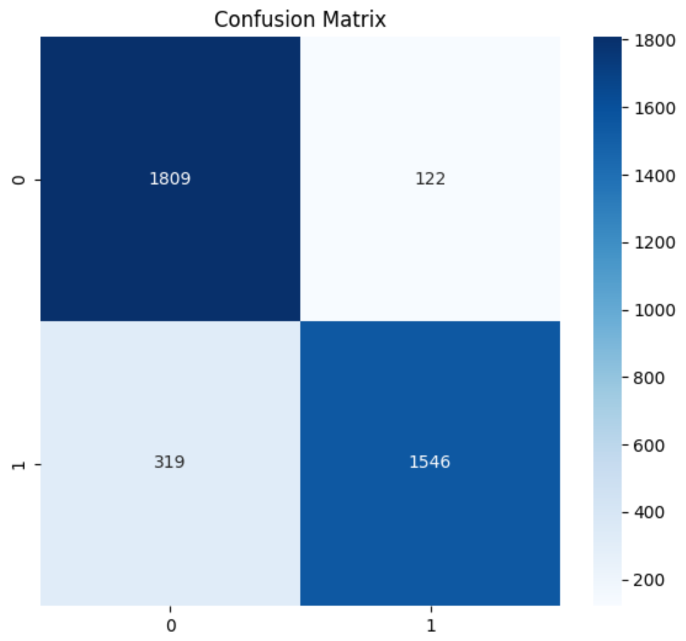
[5/7] Results and Discussion – Challenge 1 (1/4)

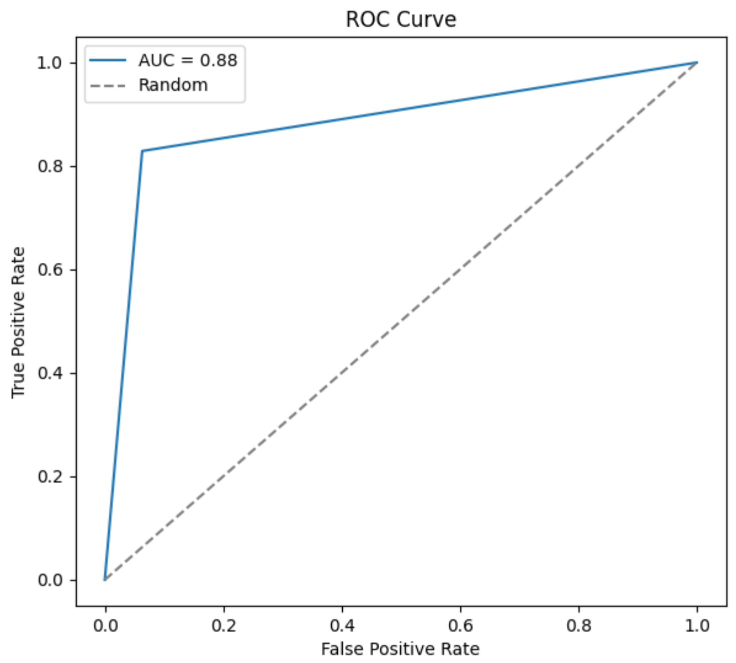
Single *(Base)* model results, trained on training split only.

Config: Preprocessing + (Hair + overall) Augmentation + Masks.

- Accuracy: 88.3825
- AUC: 88.2887
- Kappa: 76.7159

- Target 0: Sensitivity: 93.6820
- Target 1: Sensitivity: 82.8954





[5/7] Results and Discussion – Challenge 1 (2/4)

Ensemble model results trained on k=5 splits. Results only using top 3 accuracy models (slight improvement than using all 5)*.

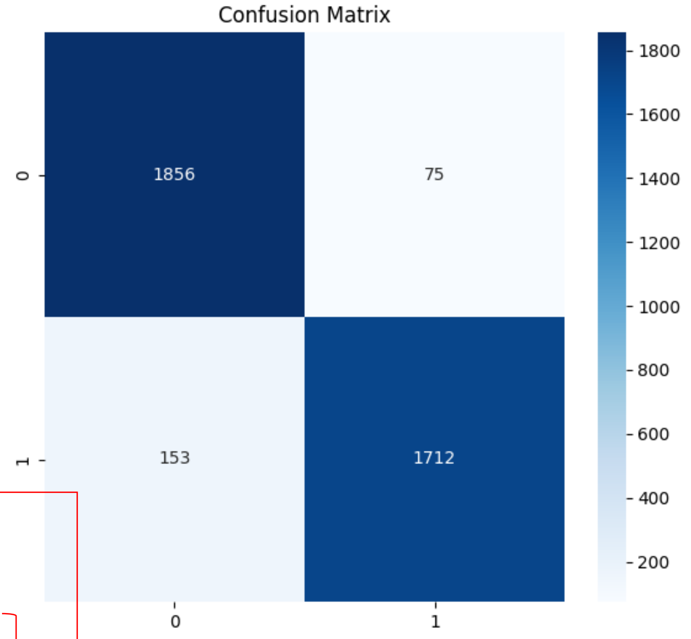
Config: Preprocessing + (Hair + overall) Augmentation + Masks.

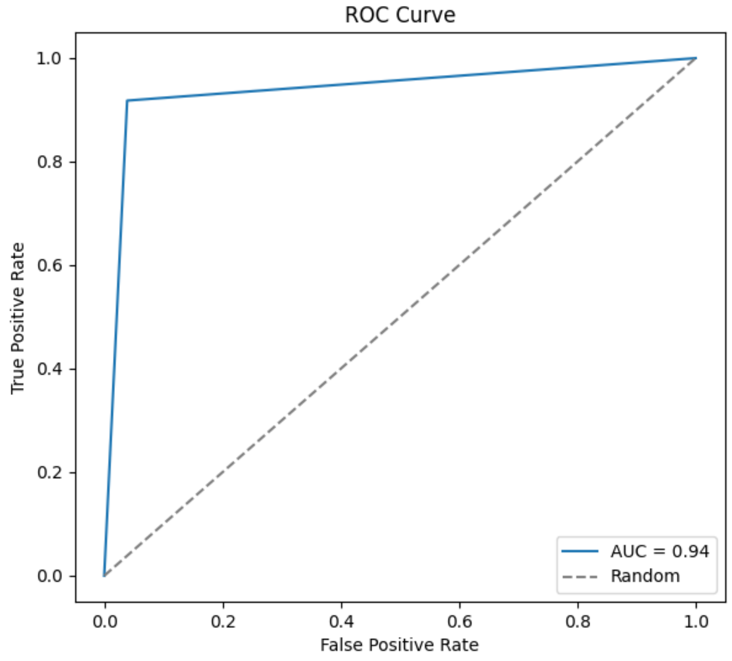
- Accuracy: 93.9937 **+ 5.6112**
- AUC: 93.9561 **+ 5.6674**
- Kappa: 87.9751 **+ 11.2492**

- Target 0: Sensitivity: 96.1160 **+ 2.4340**
- Target 1: Sensitivity: 91.7962 **+ 8.9008**

improvement made by the ensemble. ***Used for test.***

- Combination Strategy: Majority Voting
- Training time: 5 days, 12 hours





** Results obtained from all 5 models on the validation can be found in the appendix.*

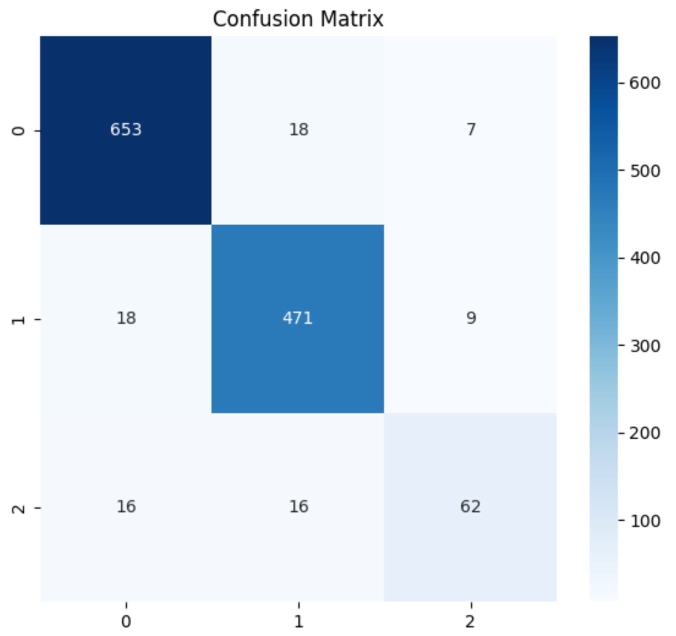
[5/7] Results and Discussion – Challenge 2 (3/4)

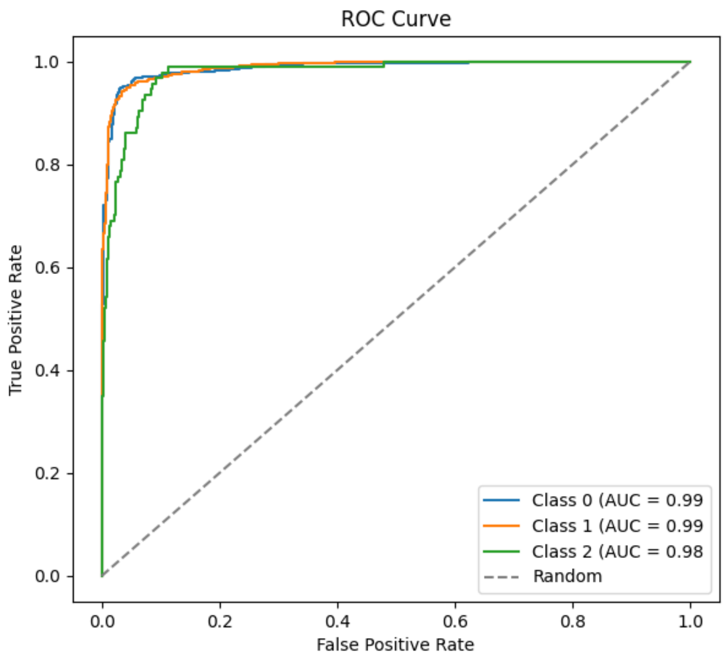
Single *(Base)* model results, trained on training split only.

Config: Preprocessing + (Hair + overall) Augmentation + Masks.

- Accuracy: 93.3858
- AUC: 98.4695
- Kappa: 87.9904

- Target 0: Sensitivity: 96.3127
- Target 1: Sensitivity: 94.5783
- Target 2: Sensitivity: 65.9574





[5/7] Results and Discussion – Challenge 2 (4/4)

Ensemble model results trained on k=5 splits. Results only using top 3 accuracy models (slight improvement than using all 5)*.

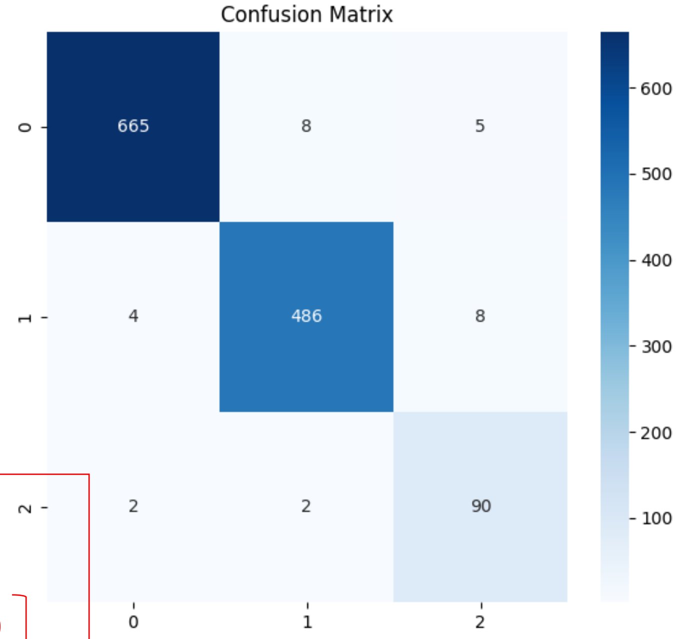
Config: Preprocessing + (Hair + overall) Augmentation + Masks.

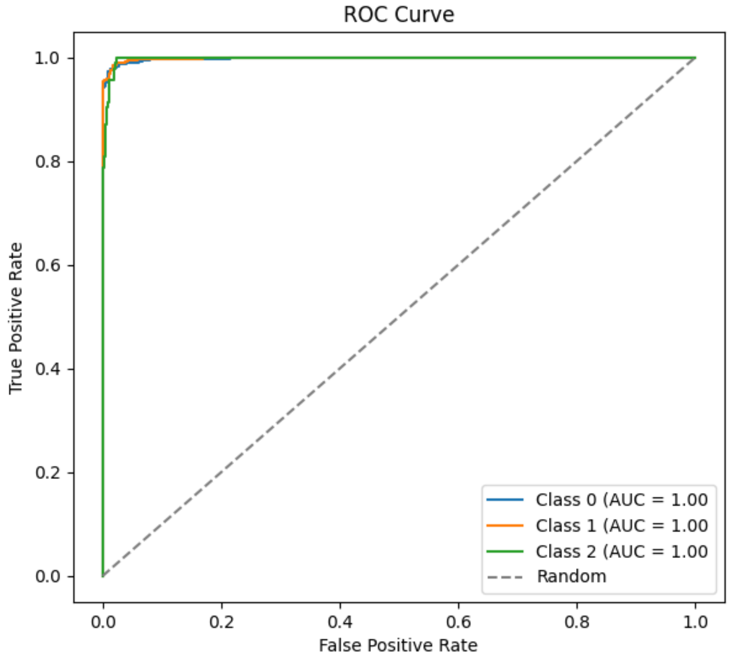
- Accuracy: 97.7165 **+ 4.3307**
- AUC: 99.8283 **+ 1.3588**
- Kappa: 95.9136 **+ 7.9232**

- Target 0: Sensitivity: 98.0826 **+ 1.7699**
- Target 1: Sensitivity: 97.5904 **+ 3.0121**
- Target 2: Sensitivity: 95.7447 **+ 29.7873**

improvement made by the ensemble. ***Used for test.***

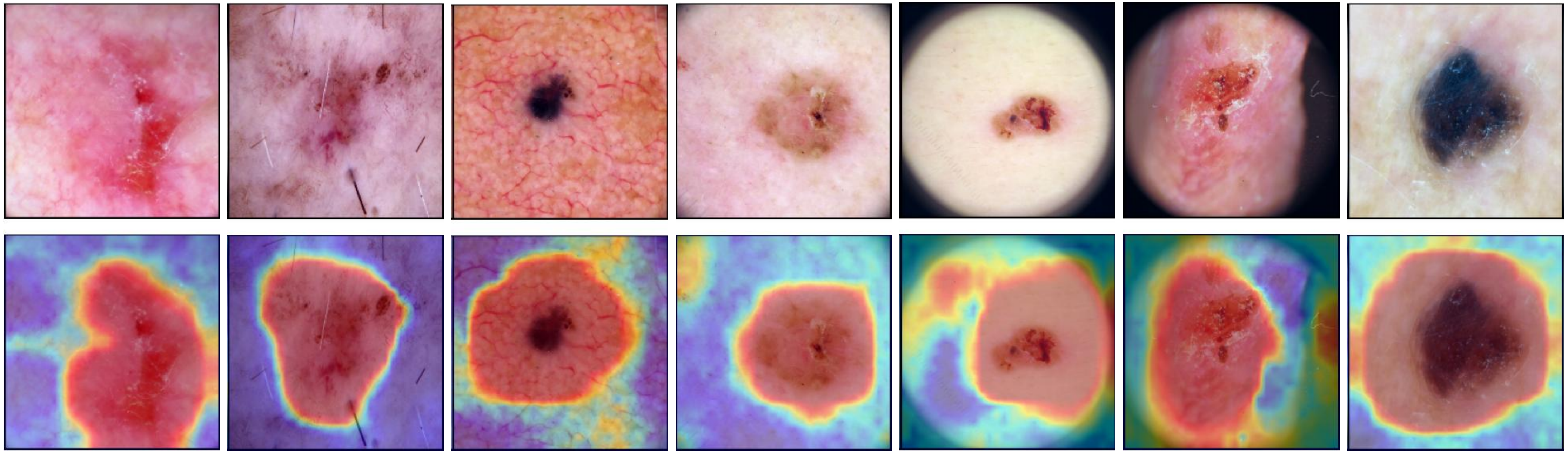
- Combination Strategy: Majority Voting
- Training time: 3 days, 9 hours





** Results obtained from all 5 models on the validation can be found in the appendix.*

[6/7] Grad-CAM Visualization



Model Fold = 3, Challenge = 2, Split = Validation, Attention Map = 1 (pooling layer 3)

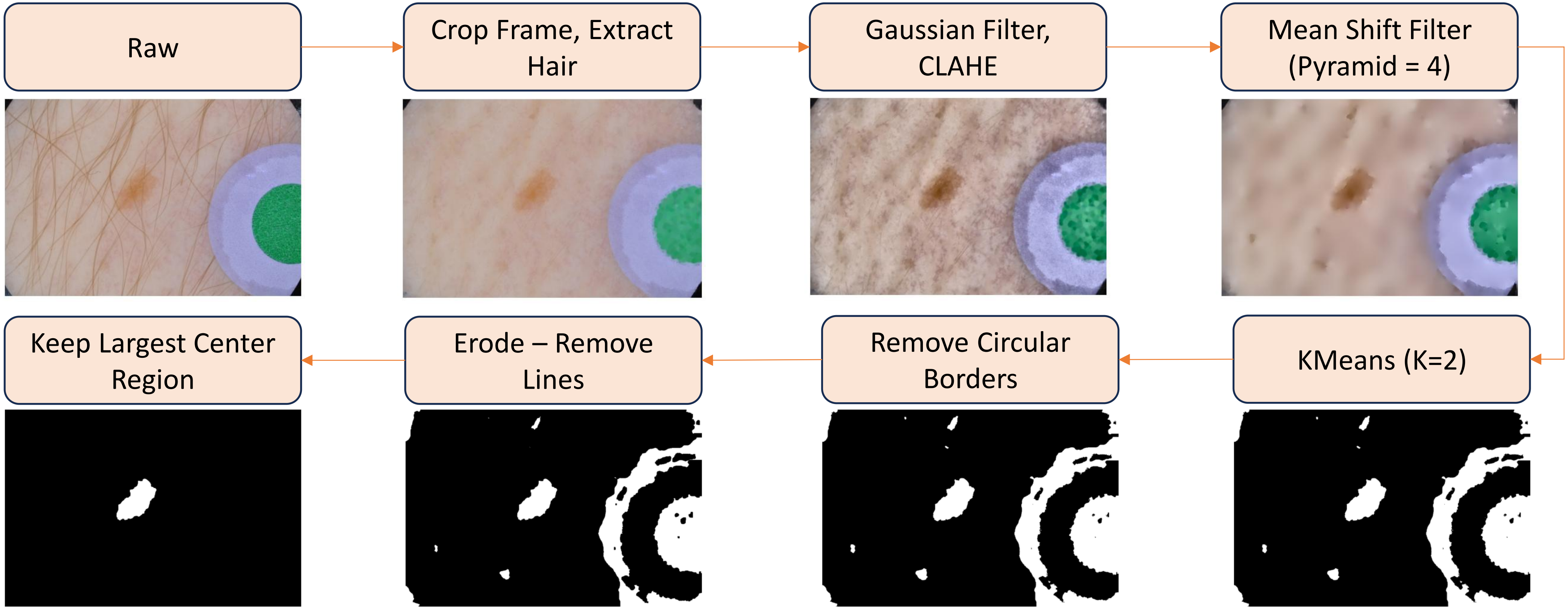
[7/7] Conclusion

- **Ensemble** approach improves the results significantly, given the same (best) model is trained on different splits of the dataset.
- Useful approaches to handle **class imbalance**:
  - StratifiedKFold (with ensemble).
  - Multi-class focal loss.
  - Augmentation.
- **Visual attention** blocks + **segmentation masks** improved slightly the results, by contributing to the training loss.
- **Augmenting the hair** slightly increased the results, making the hair challenge not focused on a specific group of images, and avoiding inpainting all the dataset.

Skin Hair Dataset (Drive Download): https://drive.google.com/drive/u/1/folders/1eNltv3y3yea7wx6mPUag8VvydgJss_mb

Code (GitHub Repository): https://github.com/abdel-habib/ISIC2019-skin-lesion-classification-segmentation

[Appendix] Skin Lesion Segmentation



Most of the lesions are in the center of the frame.

Helps removing remaining hair

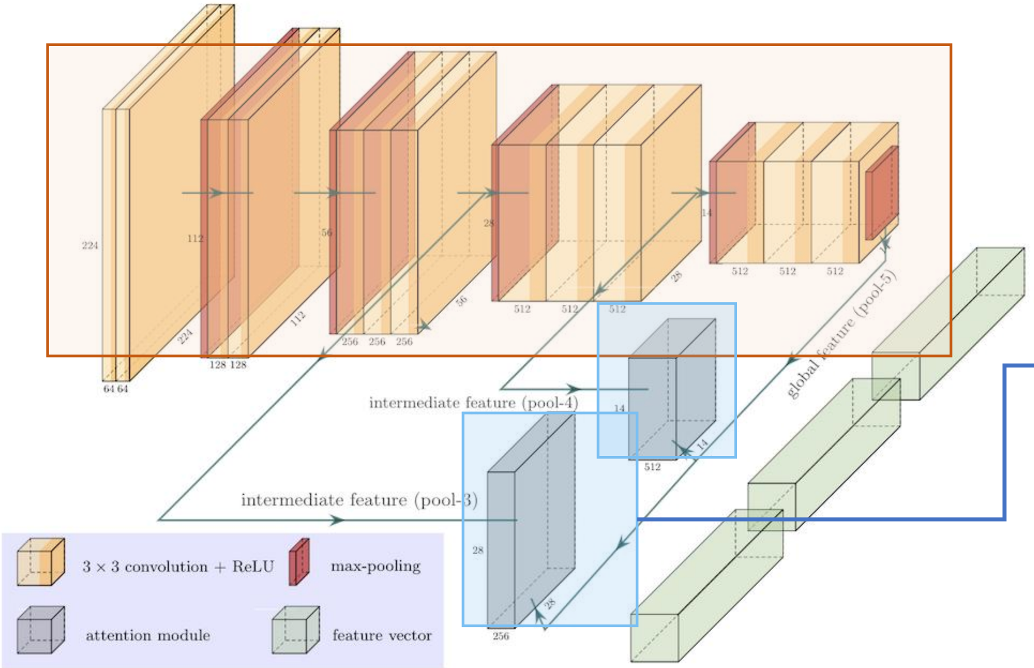
Removes microscopic corners (if not cropped)

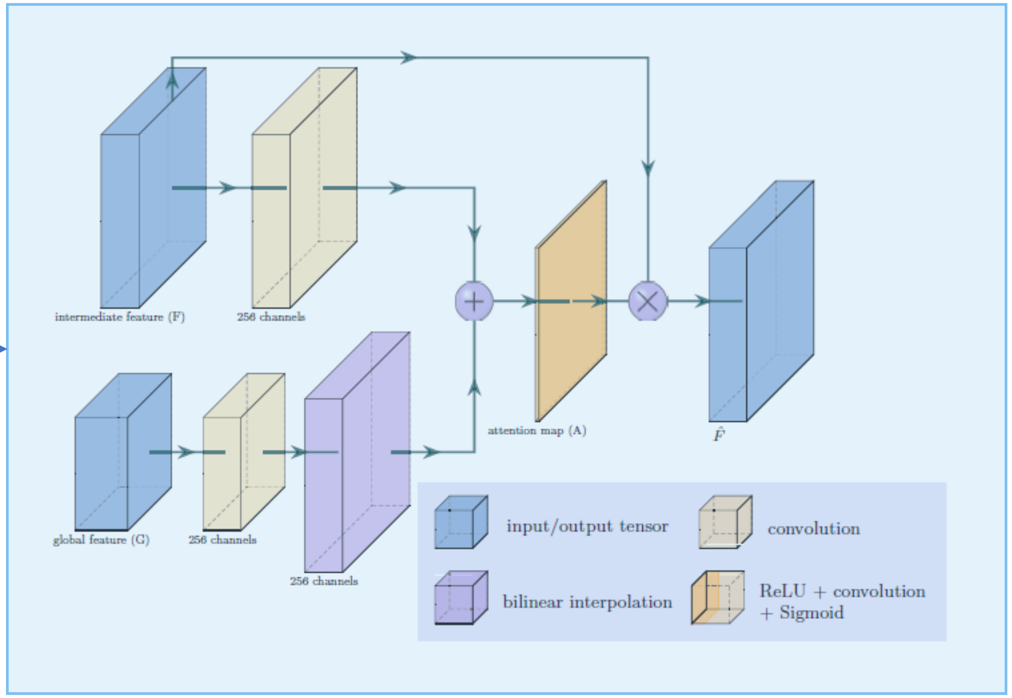
[Appendix] Architectures (Base)

Outputs: features (that we combine) + attention masks (that we use with our segmentation in the loss term)

Attention Blocks X 2

VGG16_BN Model





Yan, Y., Kawahara, J., & Hamarneh, G. (2019). Melanoma recognition via visual attention. In Information Processing in Medical Imaging: 26th International Conference, IPMI 2019, Hong Kong, China, June 2–7, 2019, Proceedings 26 (pp. 793-804). Springer International Publishing.

[Appendix] Results and Discussion – Challenge 1

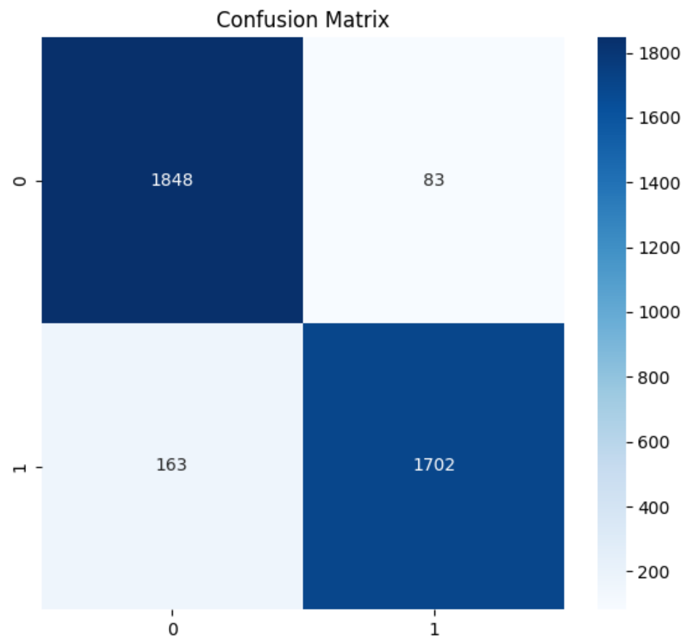
Ensemble model results, trained using k=5 folds **(using all 5 models).**

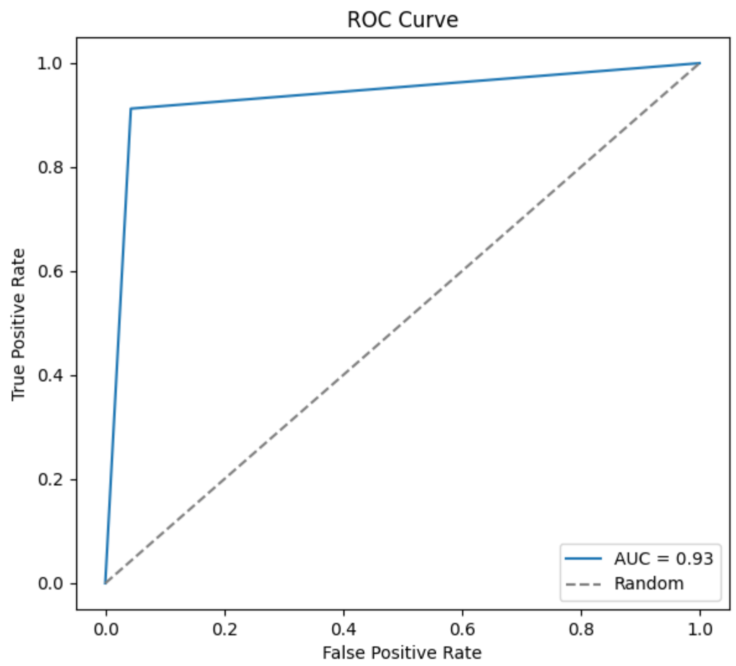
Config: Preprocessing + (Hair + overall) Augmentation + Masks.

- Accuracy: 93.5194 **+ 5.1369**
- AUC: 93.481 **+ 5.1923**
- Kappa: 87.0255 **+ 10.3096**

- Target 0: Sensitivity: 95.7017 **+ 2.0197**
- Target 1: Sensitivity: 91.2600 **+ 8.3646**

- Combination Strategy: Majority Voting
- Training time: 5 days, 12 hours





+ number – improvement made by the ensemble.

[Appendix] Results and Discussion – Challenge 2

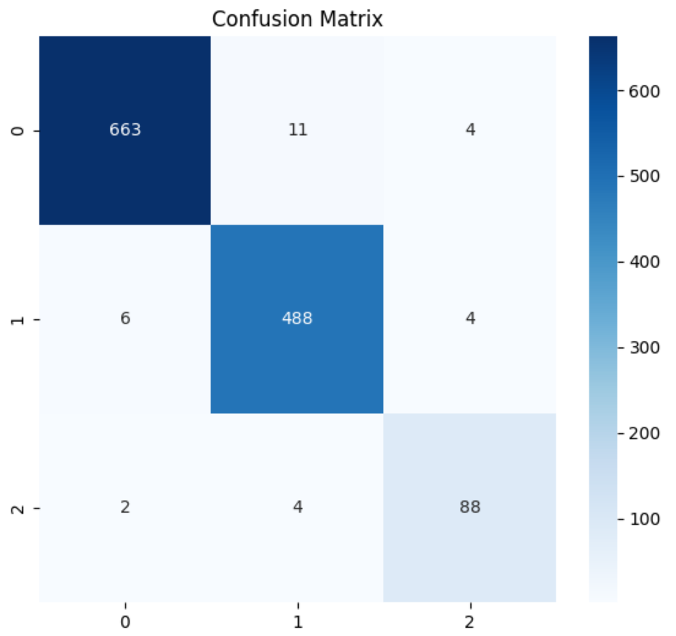
Ensemble model results, trained using k=5 folds **(using all 5 models).**

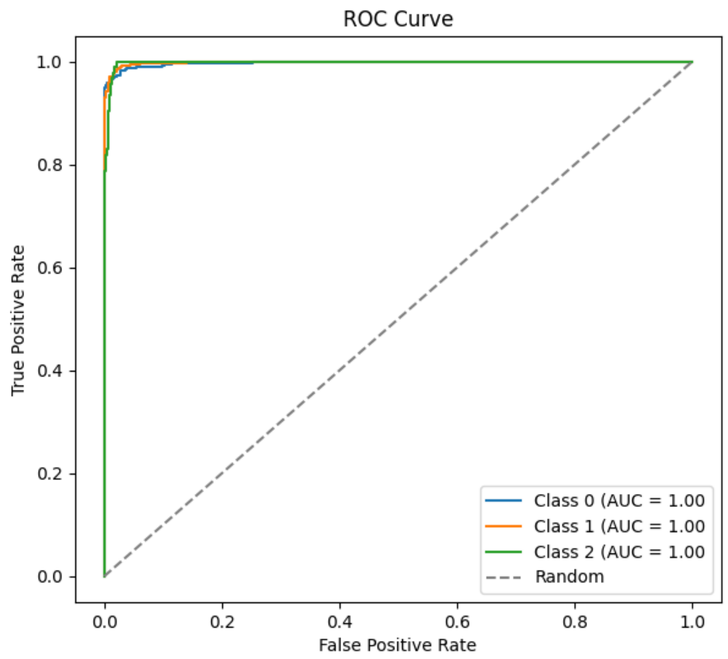
Config: Preprocessing + (Hair + overall) Augmentation + Masks.

- Accuracy: 97.5591 **+ 4.1733**
- AUC: 99.8111 **+ 1.3416**
- Kappa: 95.6179 **+ 7.6275**

- Target 0: Sensitivity: 97.7876 **+ 1.4749**
- Target 1: Sensitivity: 97.9919 **+ 3.4136**
- Target 2: Sensitivity: 93.6170 **+ 27.6596**

- Combination Strategy: Majority Voting
- Training time: 3 days, 9 hours





+ number – improvement made by the ensemble.