universidade de aveiro
departamento de eletrónica,
telecomunicações e informática

Deep learning for multi-class skin lesion diagnosis

Fábio Miguel Tomaz dos Santos

Supervisors: Filipe Silva, Pétia Georgieva

21/07/2020

Structure

1. **Motivation and Objectives**
2. CNNs and Transfer Learning
3. Experimental Setup
4. Pre-trained Model Choice and Hyperparameter Tuning
5. Improving the Model’s Generalization Performance
6. Results Discussion and Comparison
7. Conclusion

Background and Motivation

- The **International Skin Imaging Collaboration** challenges provides a **benchmark** for comparison of skin lesion classifiers:
  - Yearly challenges since 2016
  - The problem is not solved!
- **Convolutional Neural Network (CNN) models** present state-of-the-art performance in these type of challenges
- **Several Limitations** prevent these models from achieving better generalization performance:
  - Data limitations
  - Hardware limitations
  - Divergent train and test set distributions

Objectives

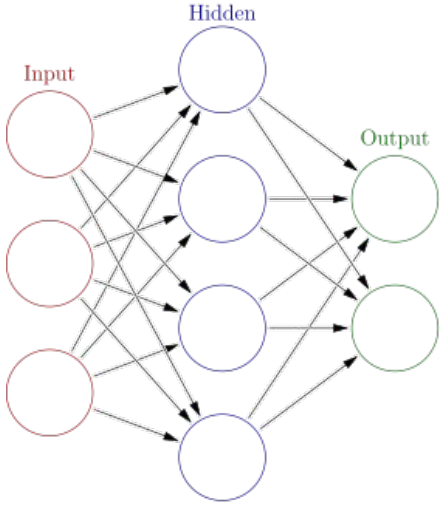
- To respond to the **ISIC 2019 challenge** by studying methods to **improve the generalization performance** of classifiers of multiple types of skin lesions
- To focus on **Convolutional Neural Network (CNN) architectures** and **transfer learning** as well as the impact of (8-class classifier):
  - **Data augmentation**
  - **Class balancing**
  - **Ensemble learning**
- To compare different strategies to identify **out of training distribution samples** from the test set (9-class classifier)

Structure

1. Motivation and Objectives
2. **CNNs and Transfer Learning**
3. Experimental Setup
4. Pre-trained Model Choice and Hyperparameter Tuning
5. Improving the Model’s Generalization Performance
6. Results Discussion and Comparison
7. Conclusion

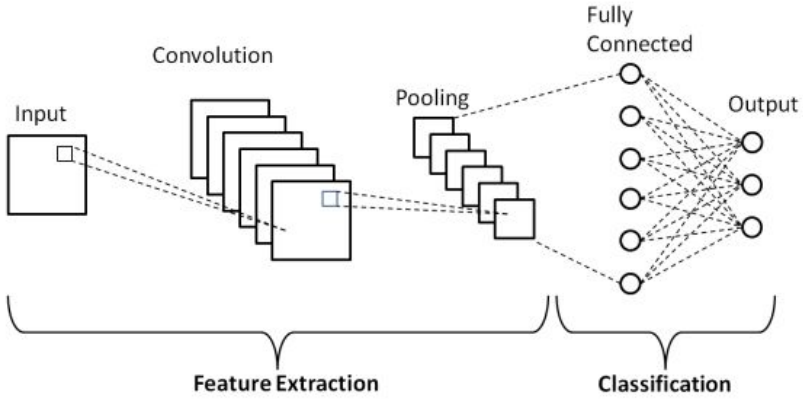
Artificial Neural Networks and Deep Learning

- Artificial Neural Network (ANN):
  - **Input layer, hidden layer, and output layer**
  - Layers composed by **neurons**
  - Activation functions (e.g., ReLU, softmax)
  - **Optimization strategy** to minimize the loss (*e.g.*, SGD, Adam)
  - **Loss function** (*e.g.*, cross entropy loss)
- Deep neural network:
  - More hidden layers
  - Can build levels of abstraction
  - **No need for feature engineering**



Convolutional Neural Networks (CNNs)

- Deep learning topology
- Used for image classification problems
- Structure:
  - Convolutional base
  - Classifier
- Distinguishing concepts:
  - **Local receptive fields** - local features
  - **Shared weights** - translation invariance
  - **Pooling layers** - information summary



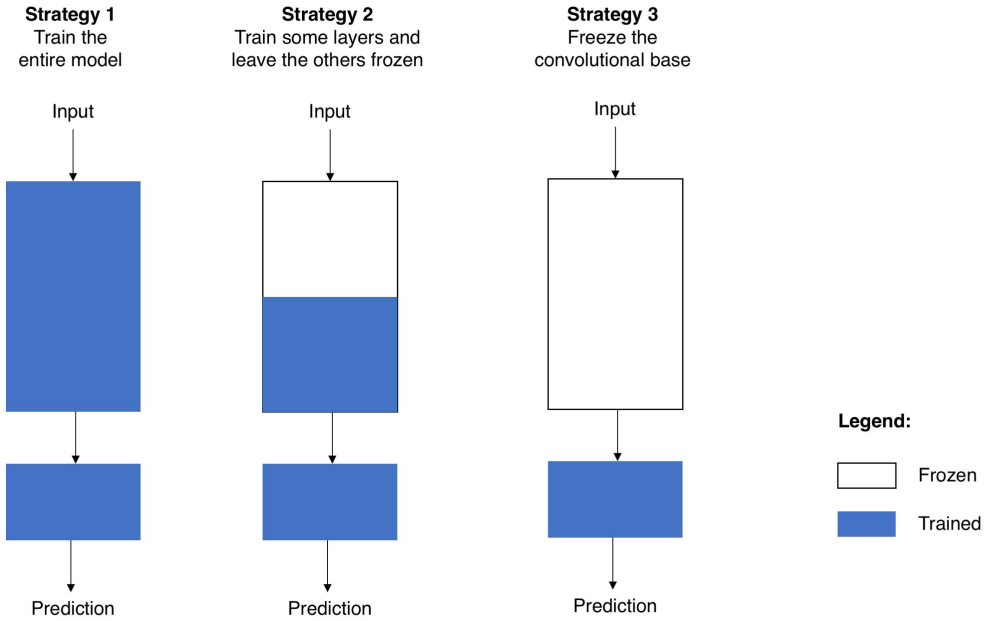
CNN Architectures

- **AlexNet:** CNN breakthrough
- **VGG:** representational depth leap
- **ResNet:** solved the vanishing gradients problem
- **DenseNet:** extended ResNet concepts
- **Inception:** Highly engineered architecture
- **EfficientNet:** Compound scaling method

Model	Year	Size	Top-1 Accuracy	Top-5 Accuracy	Params (Millions)	Depth	Input Size
AlexNet	2012	238 MB	0.570	0.803	≈ 60	8	256x256
VGG16	2014	528 MB	0.713	0.901	≈ 138	16	224x224
VGG19	2014	549 MB	0.713	0.900	≈ 143	19	224x224
ResNet50	2015	98 MB	0.749	0.921	≈ 26	50	224x224
ResNet101	2015	171 MB	0.764	0.928	≈ 45	101	224x224
ResNet152	2015	232 MB	0.766	0.931	≈ 60	152	224x224
DenseNet121	2016	33 MB	0.750	0.923	≈ 8	121	224x224
DenseNet169	2016	57 MB	0.762	0.932	≈ 14	169	224x224
DenseNet201	2016	80 MB	0.773	0.936	≈ 20	201	224x224
InceptionV3	2015	92 MB	0.779	0.937	≈ 24	159	299x299
InceptionResNetV2	2016	215 MB	0.803	0.953	≈ 56	572	299x299
EfficientNetB0	2019	5.3 MB	0.773	0.935	≈ 5	NA	224x224
EfficientNetB1	2019	7.9 MB	0.792	0.945	≈ 8	NA	240x240
EfficientNetB2	2019	9.2 MB	0.803	0.950	≈ 9	NA	260x260
EfficientNetB3	2019	12.3 MB	0.817	0.956	≈ 12	NA	300x300
EfficientNetB4	2019	19.5 MB	0.830	0.963	≈ 19	NA	380x380
EfficientNetB5	2019	30.6 MB	0.837	0.967	≈ 30	NA	456x456
EfficientNetB6	2019	43.3 MB	0.842	0.968	≈ 43	NA	456x456
EfficientNetB7	2019	66.7 MB	0.844	0.971	≈ 66	NA	600x600

Transfer Learning

- **Reusing a pre-trained model's knowledge for another related task**
- Alleviates the lack of data;
- Replace the classifier;
- Different strategies to deal with the convolutional base:
  - How large is the dataset?
  - How similar are the datasets?
  - What is the available computational capability?

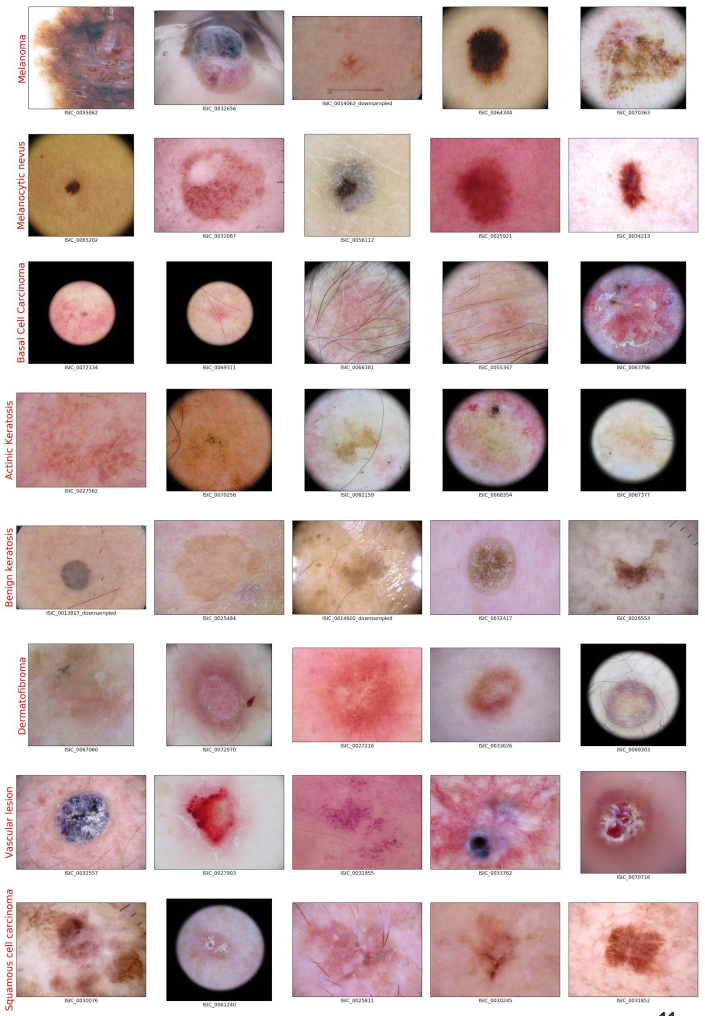


Structure

1. Motivation and Objectives
2. CNNs and Transfer Learning
3. **Experimental Setup**
4. Pre-trained Model Choice and Hyperparameter Tuning
5. Improving the Model's Generalization Performance
6. Results Discussion and Comparison
7. Conclusion

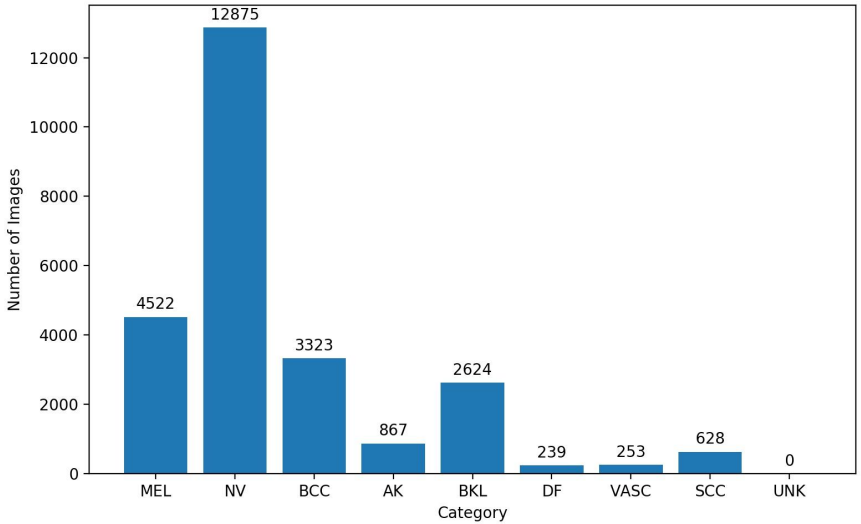
ISIC 2019 Dataset

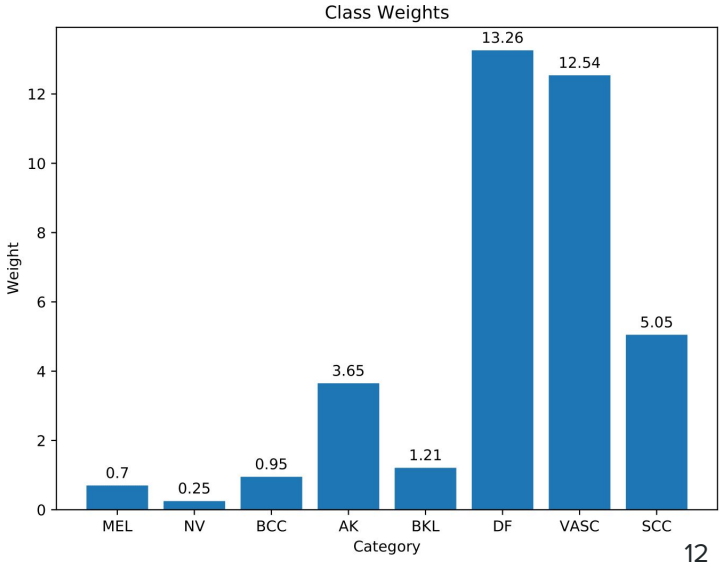
- Contains **8 classes**:
  - Melanoma (MEL)
  - Melanocytic nevus (NV)
  - Basal cell carcinoma (BCC)
  - Actinic keratosis (AK)
  - Benign keratosis (BK)
  - Dermatofibroma (DF)
  - Vascular lesion (VASC)
  - Squamous cell carcinoma (SCC)
- Pre-processing: **cropping**, **resizing**, **normalization**
- Train, validation and test split
- **Same validation & test sets across experiments**



Class Imbalance

- **High imbalance leads to poor optimization strategies**
- Solution: Assign higher loss towards underrepresented classes in the loss function



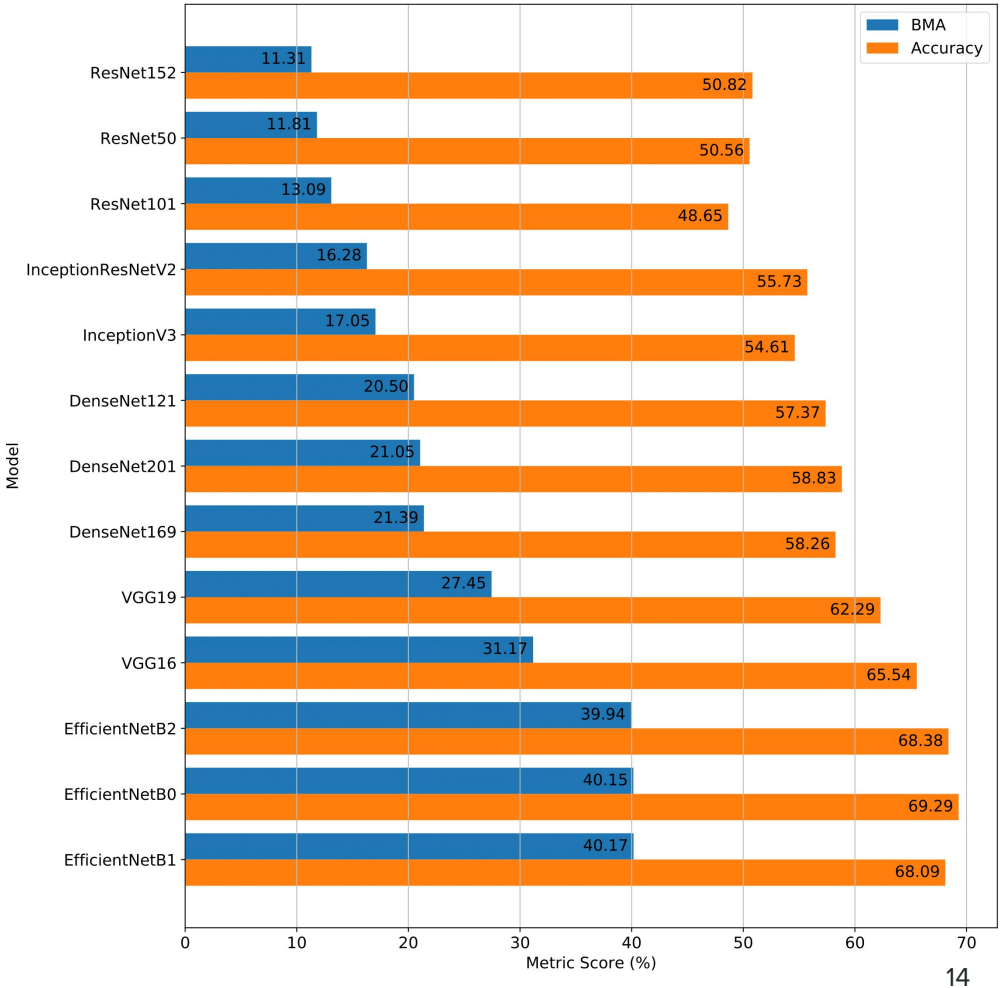


Structure

1. Motivation and Objectives
2. CNNs and Transfer Learning
3. Experimental Setup
4. **Pre-trained Model Choice and Hyperparameter Tuning**
5. Improving the Model's Generalization Performance
6. Results Discussion and Comparison
7. Conclusion

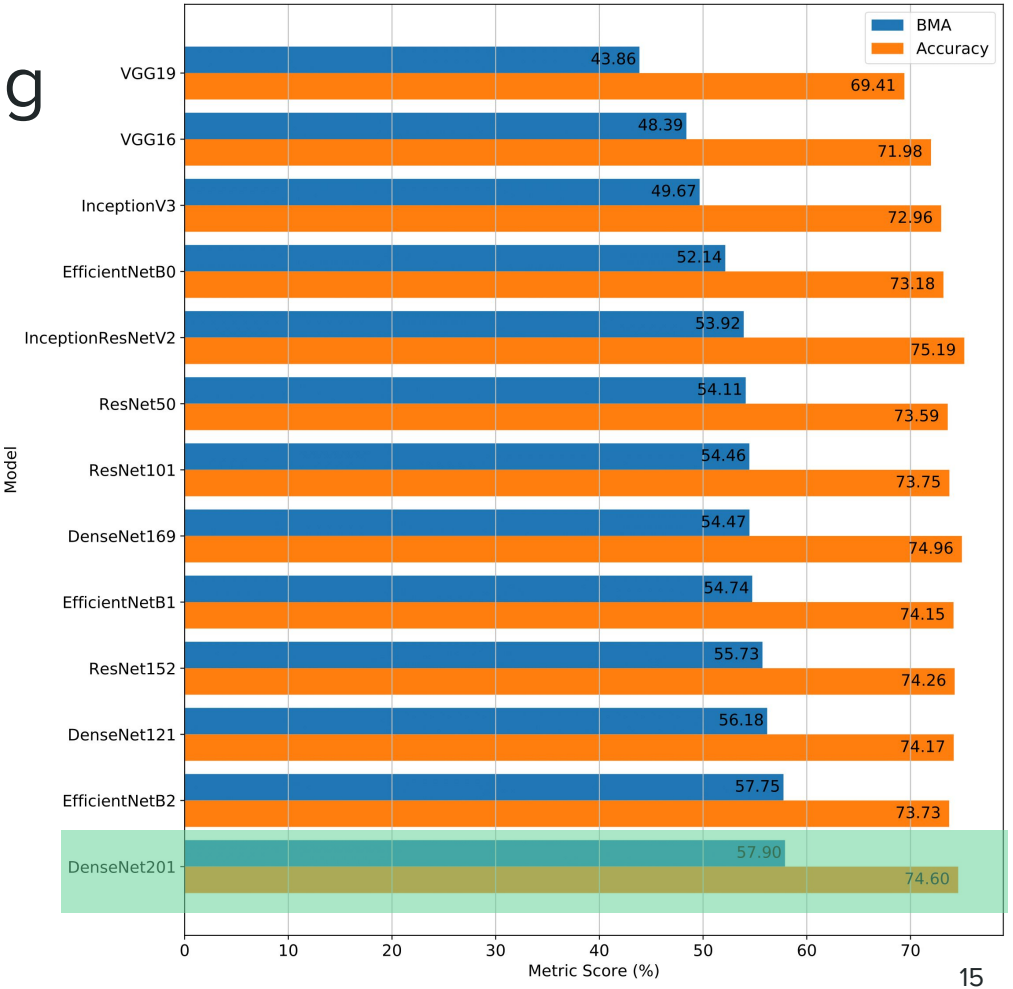
Freeze the Convolutional Base

- ImageNet is substantially different from ISIC 2019
- Convolutional base parameters are optimized for a very specific domain
- Relatively good performance on EfficientNets
- **Sub-optimal transfer learning approach for skin lesion classification!**



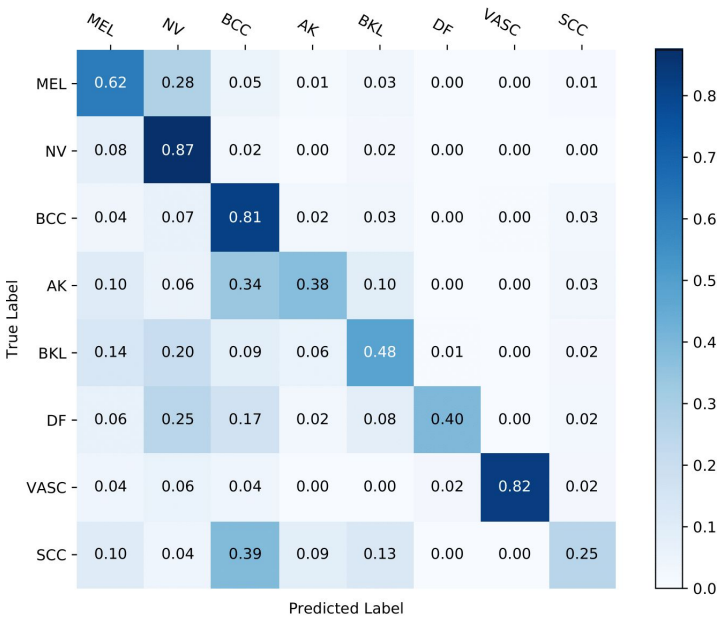
Extraction and Fine-tuning of Convolutional Layers

- **Knowledge obtained from the existing weights will be adapted to a new problem**
- **Recent architectures outperform old ones**
- Representational depth is beneficial
- Remarkable EfficientNet scalability!
- **The DenseNet201 is selected for future experiments.**

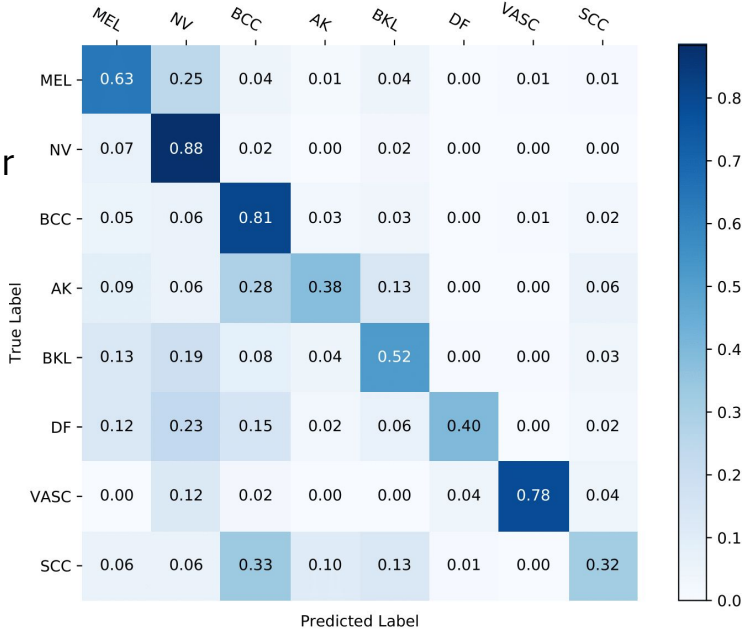


Performance Evaluation

- Poor generalization performance on underrepresented classes
- Hyperparameter optimization barely made an improvement
- **Pre-trained models are well adapted for a wide range of hyperparameters**



Hypeparameter Optimization

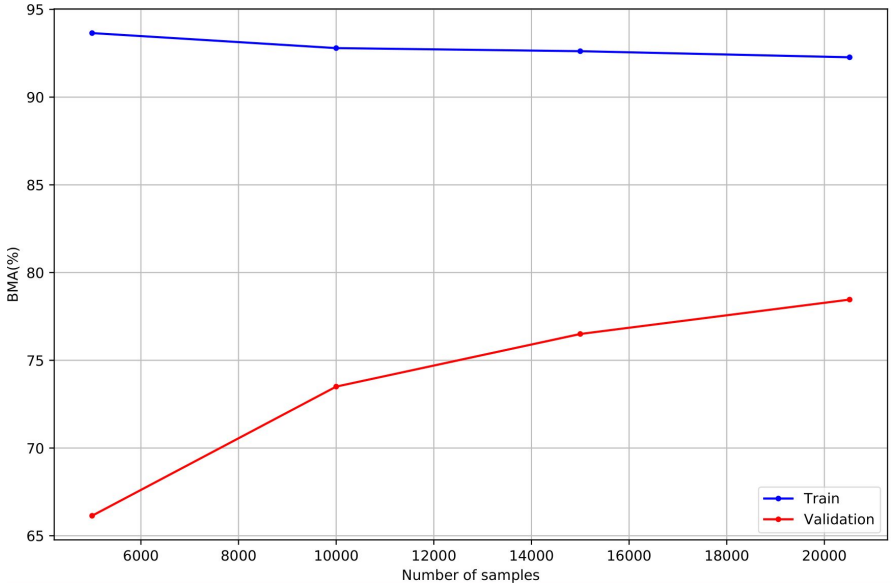


Structure

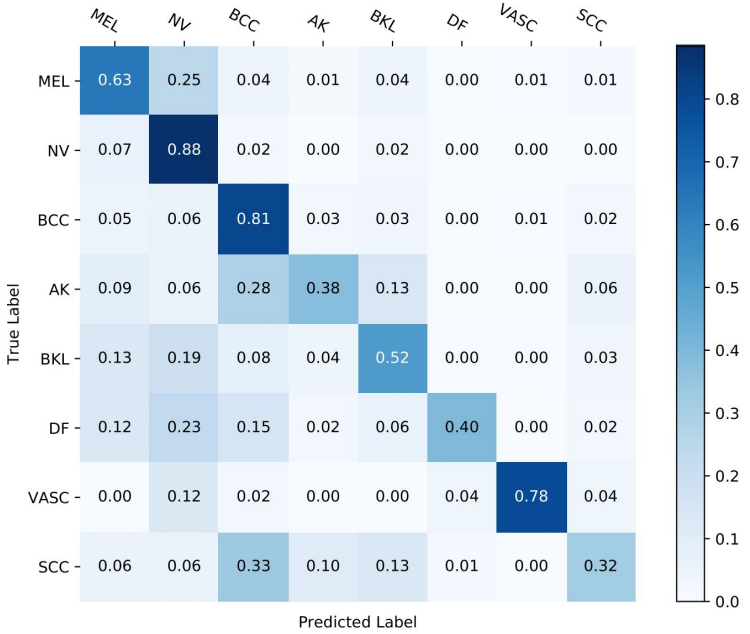
1. Motivation and Objectives
2. CNNs and Transfer Learning
3. Experimental Setup
4. Pre-trained Model Choice and Hyperparameter Tuning
5. **Improving the Model's Generalization Performance**
6. Results Discussion and Comparison
7. Conclusion

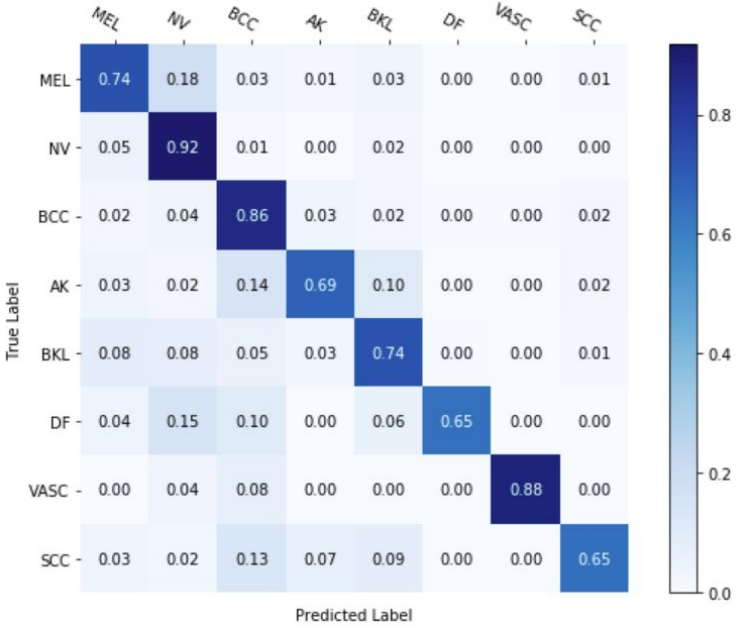
Impact of Dataset Size

- **Less overfitting**
- **Better generalization performance**
- More samples are needed to further reduce overfitting
- How to further increase performance?



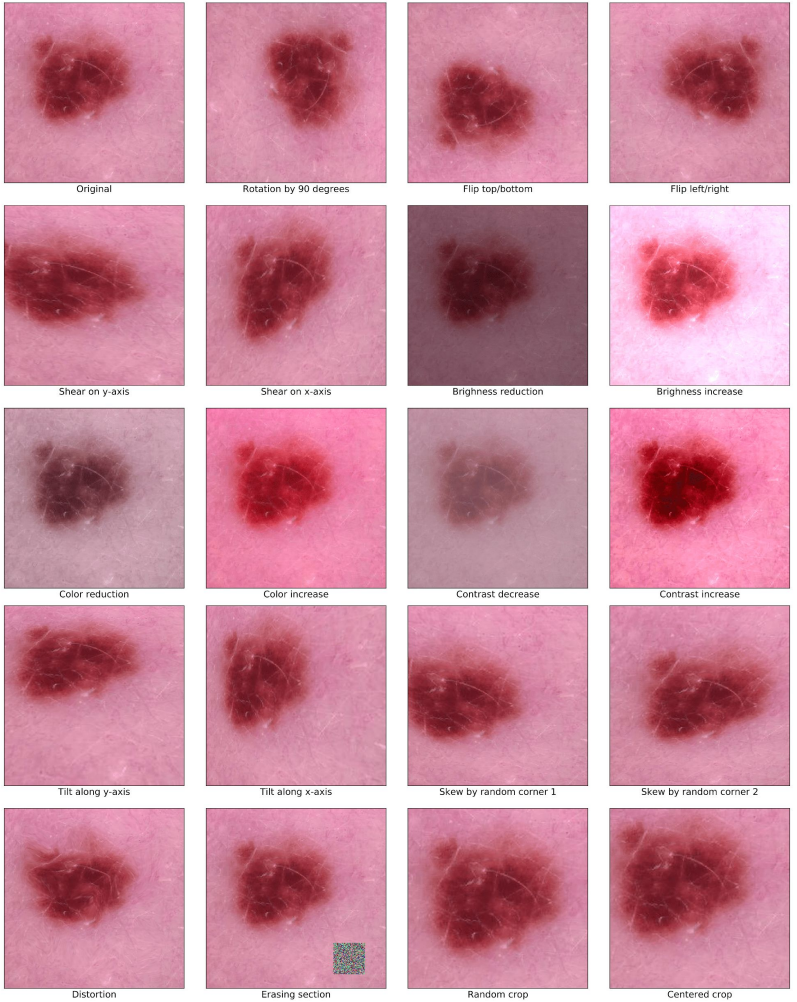
5000 samples vs 20518 samples





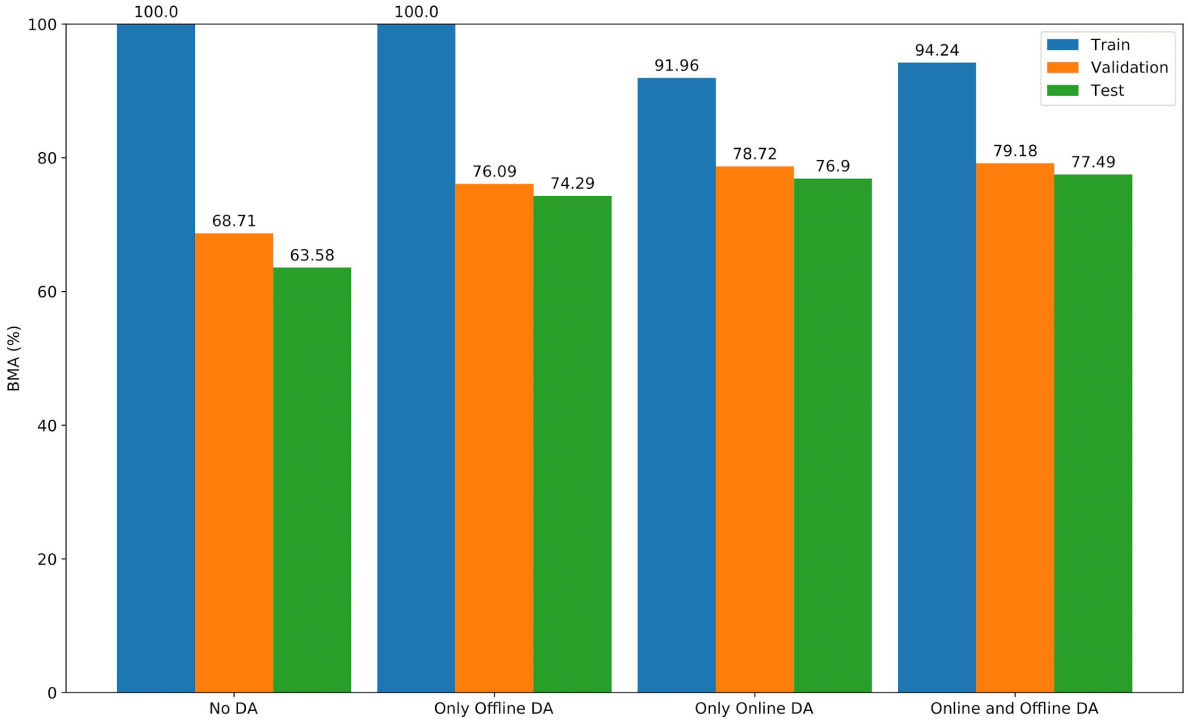
Data Augmentation Groups

- **Group 0:** No augmentation techniques
- **Group 1:** Slight variations to the original image;
- **Group 2:** Augmentation techniques of group 1 plus some adjustments on pixel intensities
- **Group 3:** Augmentation techniques of group 1 plus perspective transformations
- **Group 4:** Augmentation techniques of group 1 plus noise induction augmentation techniques



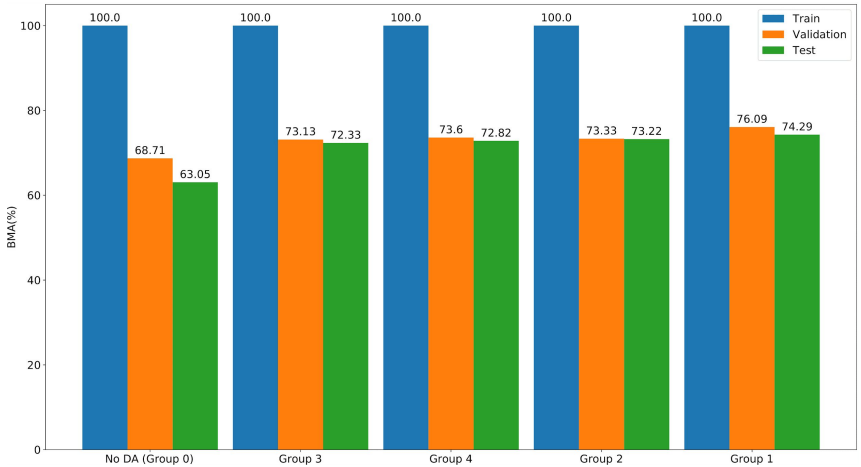
Offline vs Online Data Augmentation

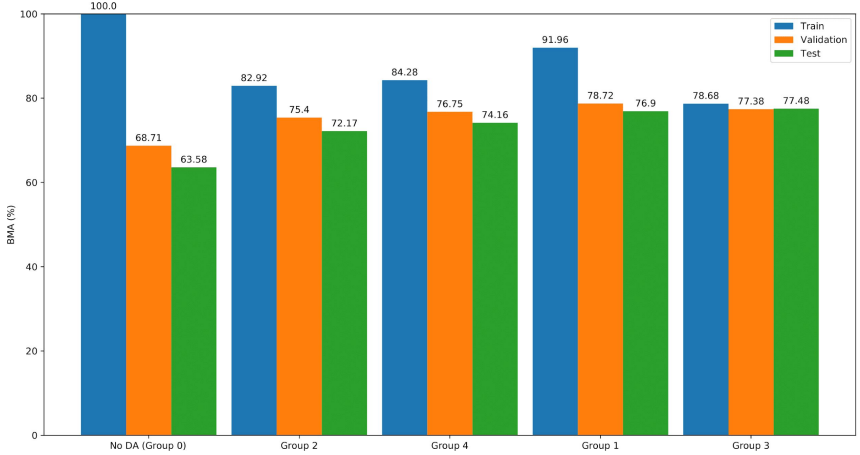
- **Offline** data augmentation used to **class balance** and **oversample** the dataset
- **Online** data augmentation used to **reduce overfitting** while training
- **Combining both reduces overfitting and improves generalization performance**



Augmentation Group Results

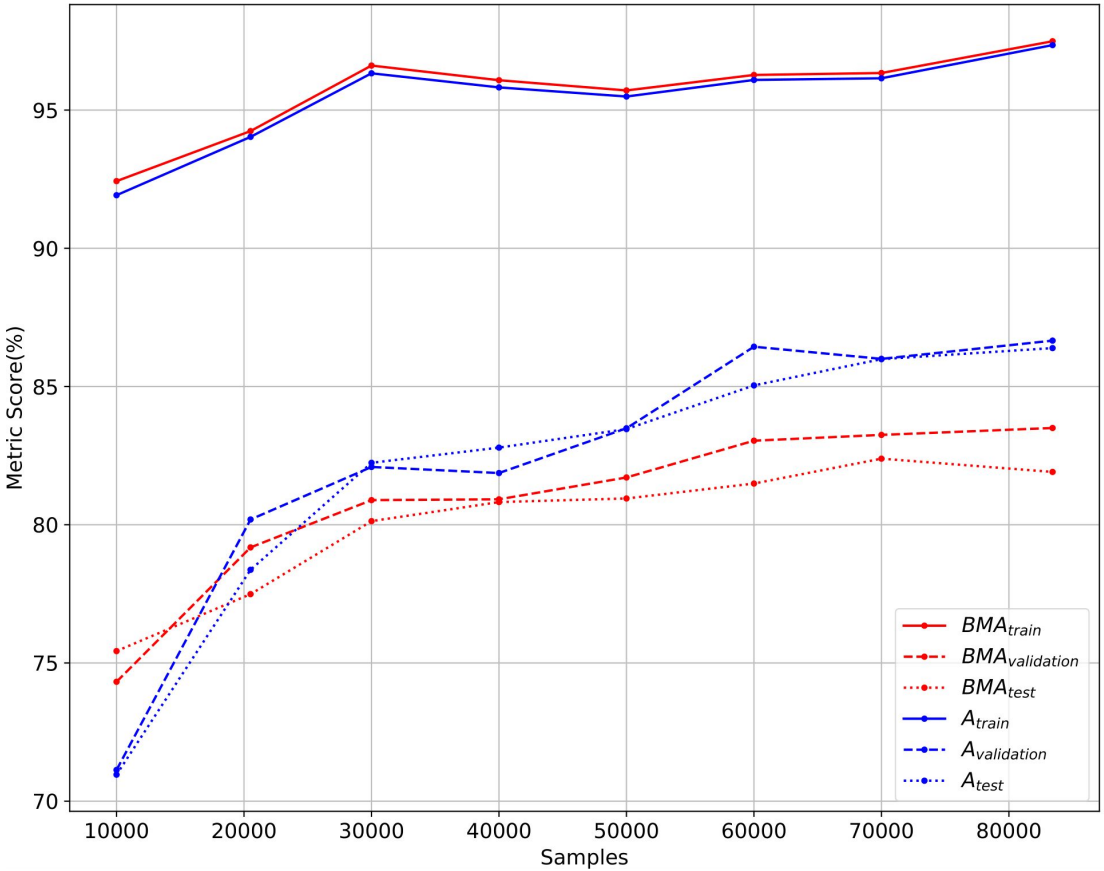
- **Offline:**
  - Complex augmentation remove important information
  - **Simpler augmentations perform better**
- **Online:**
  - Group 2 and 4 reduce overfitting but also reduce train, validation and test BMA
  - Group 3 dramatically reduces overfitting
  - Results are related to the way online data augmentation works
  - **Complex augmentations help generalize knowledge**



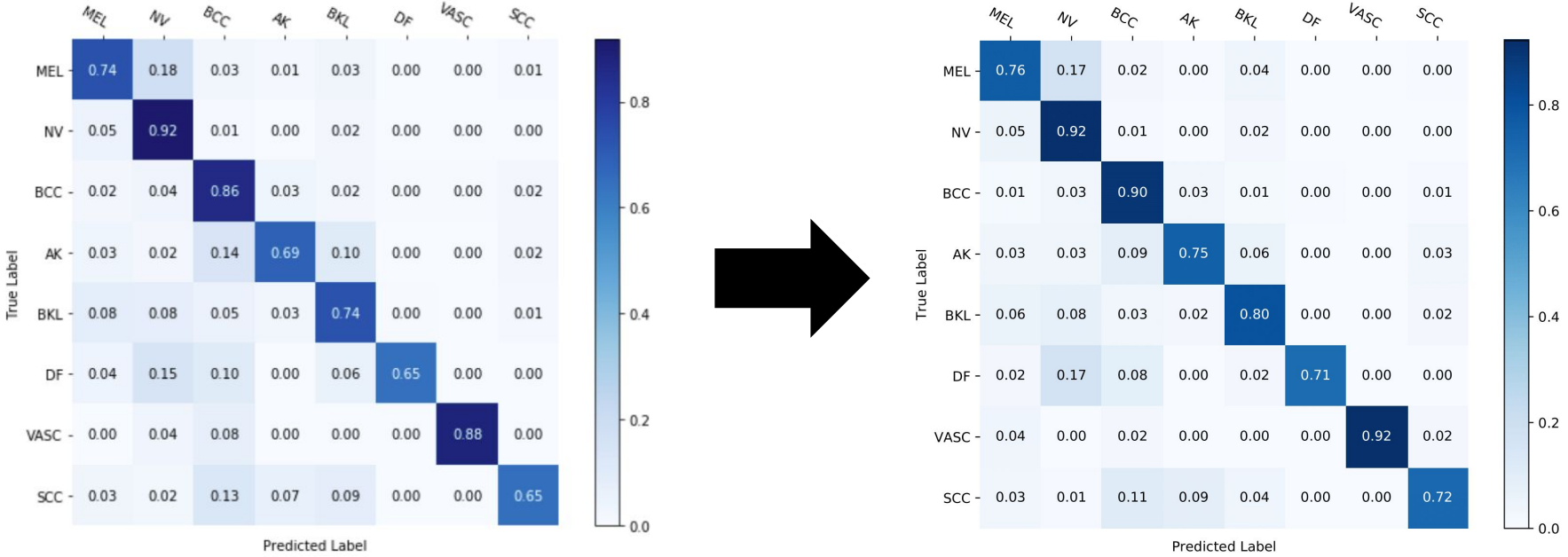


Class Balancing

- Oversampling through augmentation group 1
- **Performance increase on underrepresented classes**
- Gap between BMA and accuracy



Original Training Dataset (20518 samples) vs Synthetic Class-Balanced Dataset (83432 samples)



Ensemble

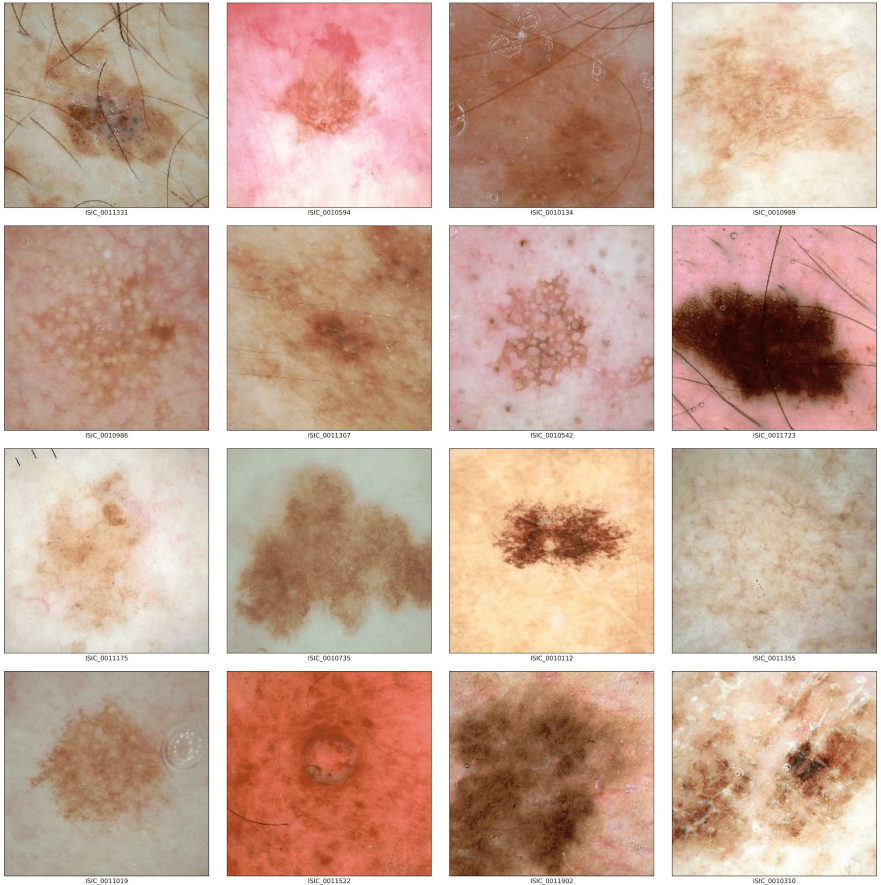
- Considerable improvement from single-model performance
- More models are not necessarily better
- **Very useful for benchmark challenges**

Approach Name	Pre-trained models	BMA	Accuracy
EfficientNetB2 approach	EfficientNetB2	≈ 0.820	≈ 0.851
DenseNet121 approach	DenseNet121	≈ 0.815	≈ 0.867
InceptionResNetV2 approach	InceptionResNetV2	≈ 0.811	≈ 0.862
ResNet152 approach	ResNet152	≈ 0.797	≈ 0.858
VGG16 approach	VGG16	≈ 0.753	≈ 0.824
Ensemble of best 2 models	EfficientNetB2, DenseNet121	≈ 0.842	≈ 0.878
Ensemble of best 3 models	EfficientNetB2, DenseNet121, InceptionResNetV2	≈ 0.846	≈ 0.891
Ensemble of best 4 models	EfficientNetB2, DenseNet121, InceptionResNetV2, ResNet152	≈ 0.841	≈ 0.894
Ensemble of all 5 models	EfficientNetB2, DenseNet121, InceptionResNetV2, ResNet152, VGG16	≈ 0.836	≈ 0.893

Single Model vs Ensemble of Best Three Models

Out of Training Distribution Samples

- Composed of different types of lesions from ISIC Archive, not belonging to the original distribution
- Same preliminary steps (i.e., pre-processing and split)



Out of training distribution detection

- Methods from the literature:
  - Top-1 softmax thresholding
  - Out-of-DIstribution detector for Neural networks (ODIN)
  - Outlier Class
- Outlier class outperforms other approaches
- Some limitations:
  - Low generalization performance:
    - Small set of samples
    - Clearly defined distribution
  - Bias in the training and test sets

Out-of-distribution method	BMA	Accuracy
Softmax threshold (0.7)	0.721	0.823
ODIN	0.654	0.691
Outlier class	0.750	0.847

Structure

1. Motivation and Objectives
2. CNNs and Transfer Learning
3. Experimental Setup
4. Pre-trained Model Choice and Hyperparameter Tuning
5. Improving the Model's Generalization Performance
6. **Results Discussion and Comparison**
7. Conclusion

Results Discussion

- **Important methods to improve generalization performance:**
  - Transfer learning methodology + CNN pre-trained model
  - Dataset size
  - Data augmentation
- **Methods to consider:**
  - Hyperparameter tuning
  - Ensemble learning
  - Out-of-training distribution detection methods

Approach Step	BMA	Accuracy
Pre-trained model choice (see Section 5.1)	≈ 0.579	≈ 0.746
Hyperparameter tuned model (see Section 5.2)	≈ 0.585	≈ 0.754
Full dataset model without online DA (see Section 6.1.2)	≈ 0.636	≈ 0.799
Full dataset model with online DA (see Section 6.1.2)	≈ 0.769	≈ 0.849
Class balanced model (see Section 6.1.3)	≈ 0.815	≈ 0.867
Ensemble of 3 models (see Section 6.2)	≈ 0.846	≈ 0.891
Model with out-of-distribution detection (see Section 6.3)	≈ 0.750	≈ 0.847

Results Comparison

- **State of the art results on both single-model and multi-model performance for 8-class classification**
- 9-class classification tested with a different test set

Approach	In distribution (8 Classes)		Out of distribution (9 Classes)	
	BMA_{test}	A_{test}	BMA_{test}	A_{test}
3 Model Ensemble	0.846	0.891	0.775	0.858
DenseNet121	0.815	0.867	0.750	0.847
Gessert *et al.* (2019)	0.725	NA	0.636	NA
Zhou *et al.* (2019)	0.753	NA	0.607	NA
Hsin-Wei Wang (2019)	0.828	NA	0.505	NA

Structure

1. Motivation and Objectives
2. CNNs and Transfer Learning
3. Experimental Setup
4. Pre-trained Model Choice and Hyperparameter Tuning
5. Improving the Model's Generalization Performance
6. Results Discussion and Comparison
7. **Conclusion**

Key Takeaways

- **Fine-tuning the convolutional base** is the optimal approach for skin lesion classification
- The choice of the **pre-trained model's architecture** can have a substantial impact
- Pre-trained models are well optimized for a wide range of **hyperparameters**
- The **augmentation techniques** used should be carefully selected
- **Online data augmentation** emerges as a method to reduce overfitting
- **Ensembling** multiple models is useful for benchmarks
- Re-training the model with an **outlier class** can become a viable approach as an **out of training distribution detection method**

Future Research

- Further research towards **online data augmentation**:
  - Explore other augmentation groups (*e.g.*, GANs)
  - Study the impact on other datasets (*e.g.*, ISIC 2020 challenge dataset)
- Lack of **interpretability/explainability** of deep learning models:
  - Hierarchical classifier
  - Display kernel filters during the inference phase
- Further research is required to integrate these models into the **clinical workflow**:
  - Get dermatology professionals involved
  - Focus on practicality rather than performance

Thank You