UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

RETINAL LESIONS SEGMENTATION USING CNNS AND ADVERSARIAL TRAINING

In partial fulfilment of the requirements for the degree in Telecommunications Technologies and Services Engineering

NATÀLIA GULLÓN

ADVISOR: VERÓNICA VILAPLANA

July 2018

CONTENTS

CONTENTS

INTRODUCTION

Problem statement

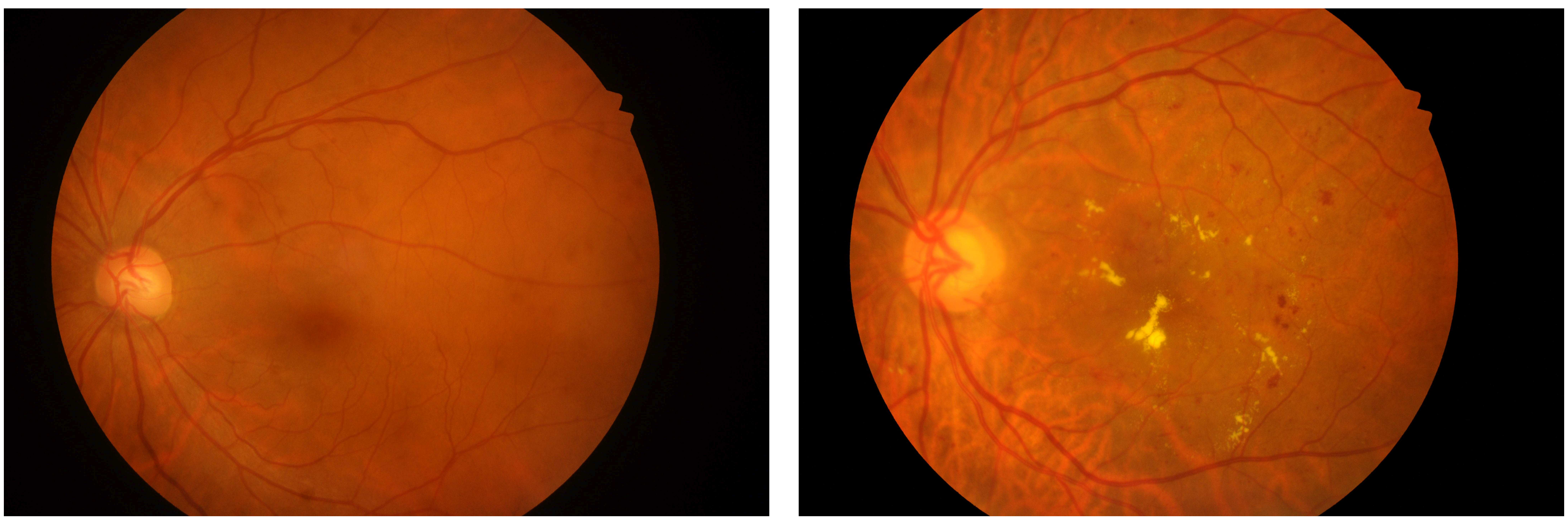
- Interpretations of medical data are done by experts
- Problem: cost of these interpretations very high
- Solution: deep learning techniques to help experts

Medical interpretations made by experts

INTRODUCTION

Diabetic retinopathy

- Leading cause of blindness among working-age adults
- It affects people with diabetes and the damage occurs in the retina



Funduscopy images. Left: healthy retina. Right: retina with lesions

INTRODUCTION

Project overview

- Objectives: apply deep learning techniques to detect and segment retinal lesions
- Convolutional Neural Networks (CNNs) will be used
- Adversarial training will be added to the previous CNN to improve its results
- Dataset: *Indian Diabetic Retinopathy Image Dataset* (IDRiD) from the *Diabetic Retinopathy: Segmentation and Grading Challenge*

CONTENTS

STATE OF THE ART

Semantic segmentation

- Convolutional Neural Networks:
    - One of the models with successful results: U-Net
- Adversarial training

Automation of Diabetic Retinopathy diagnostics

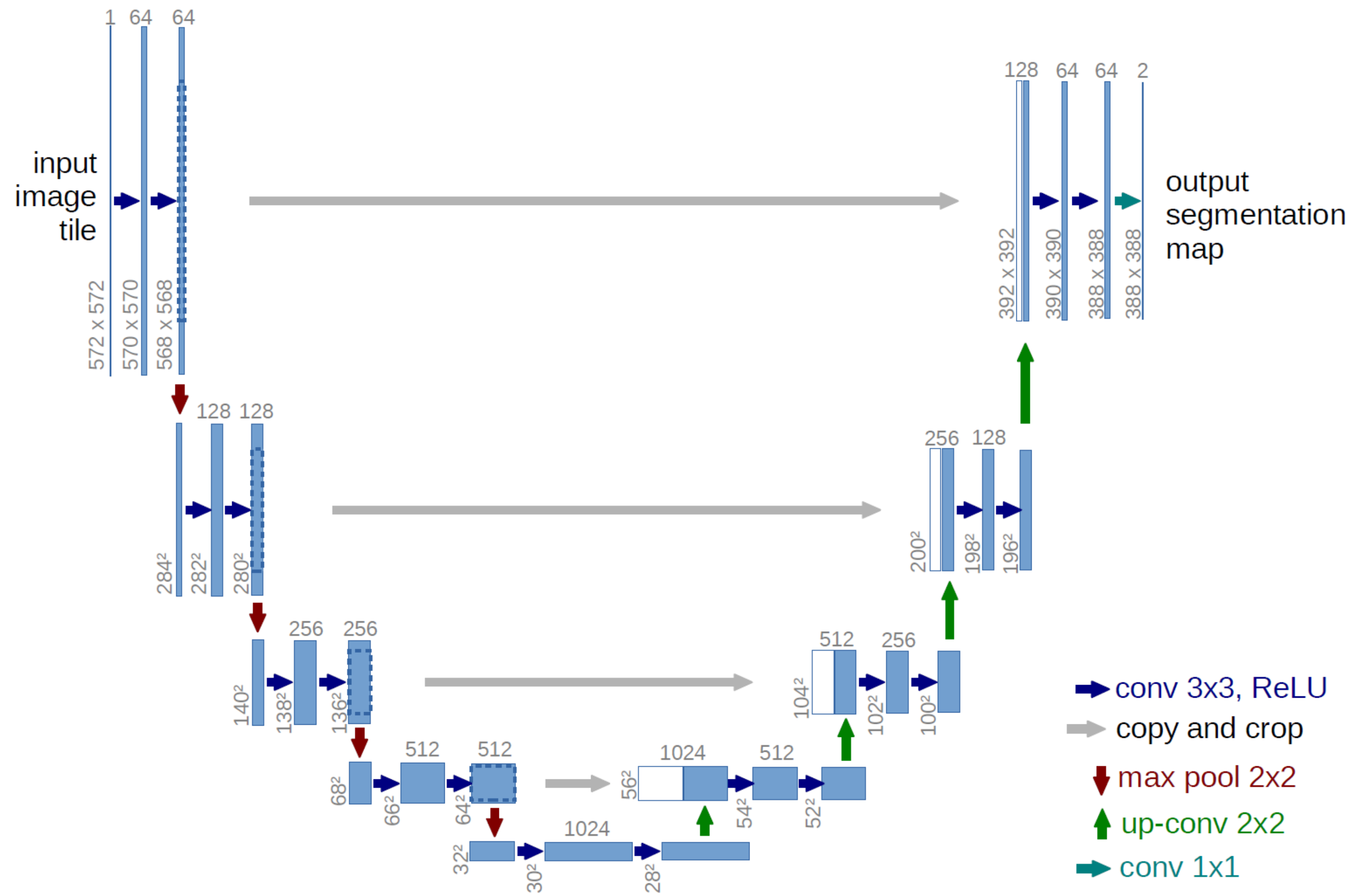
- Most work: classification into different stages of DR
- Low accuracy on retinal lesions segmentation
- Relevant work on detecting anatomical structures of blood vessels, fovea and optic disk

CONTENTS

METHODOLOGY

U-Net

- CNN used to segment images
- Structure: contracting path + expansive path

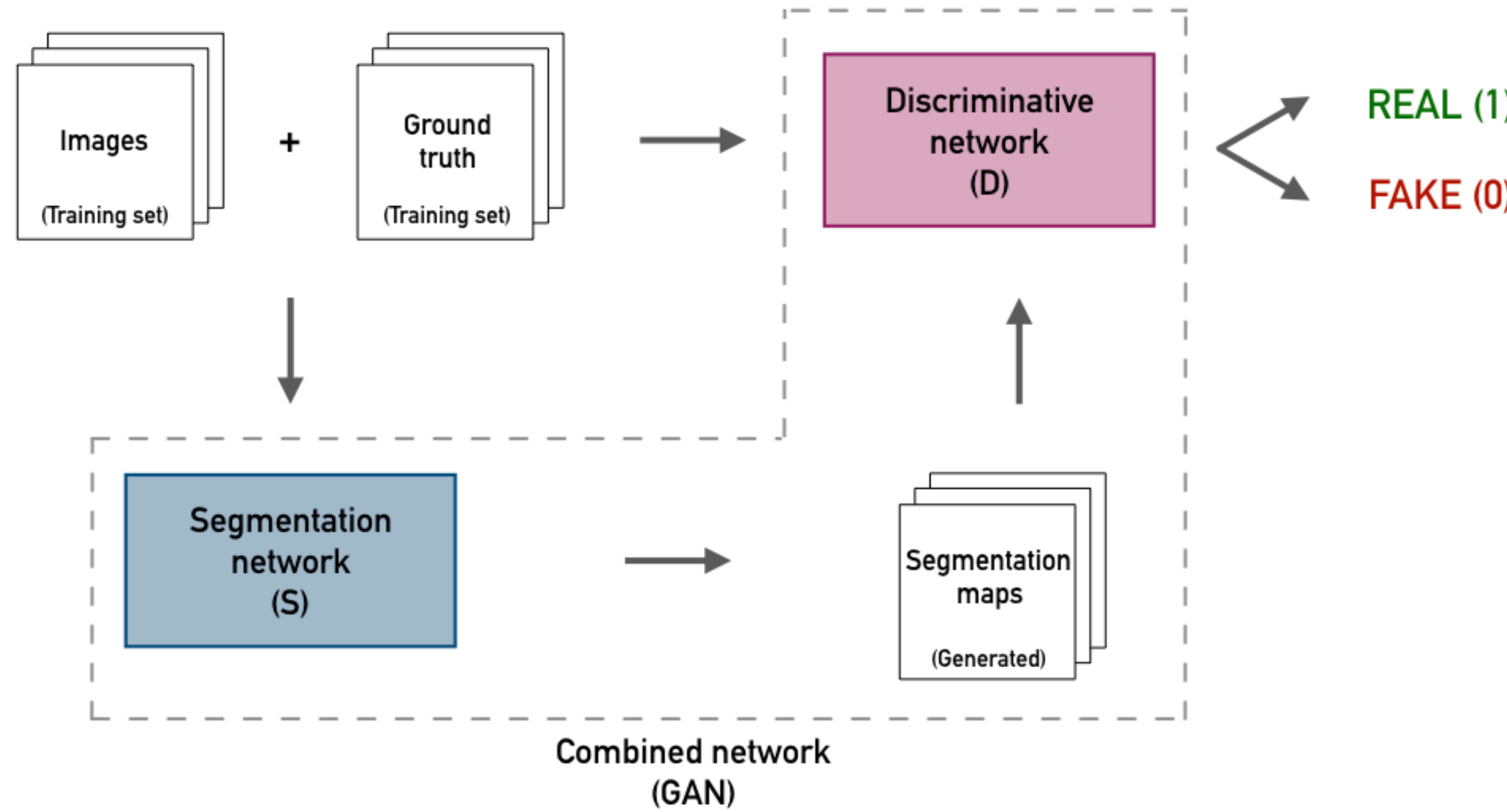


U-Net structure

METHODOLOGY

Adversarial training

- Idea used by GANs applied to train segmentation models
- Structure: two networks pitting one against the other
- Adversarial training procedure is a two-player minimax game:



Architecture used with adversarial training

$$\min_S \max_D V(S,D) = \mathbb{E}_{x,y\sim p_{data}(x,y)}[\log D(x,y)] + \mathbb{E}_{x\sim p_{data}(x)}[\log(1 - D(x, S(x)))]$$

METHODOLOGY

Loss functions

- U-Net:
  - Categorical cross-entropy
  - Dice coefficient
  - Generalised dice coefficient
- Adversarial training:
  - Discriminator:
    - Binary cross-entropy
    - Mean squared error
  - Segmentation network:
    - $L_{GAN} = \lambda \cdot L_{ADV} + L_{SEG}$

METHODOLOGY

Metrics

- To evaluate the models, we used the dice score:

$$D = \frac{2|P \cap G|}{|P| + |G|} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}$$

P: Predicted labels, G: Ground truth

TP: True Positives, FP: False Positives, FN: False Negatives

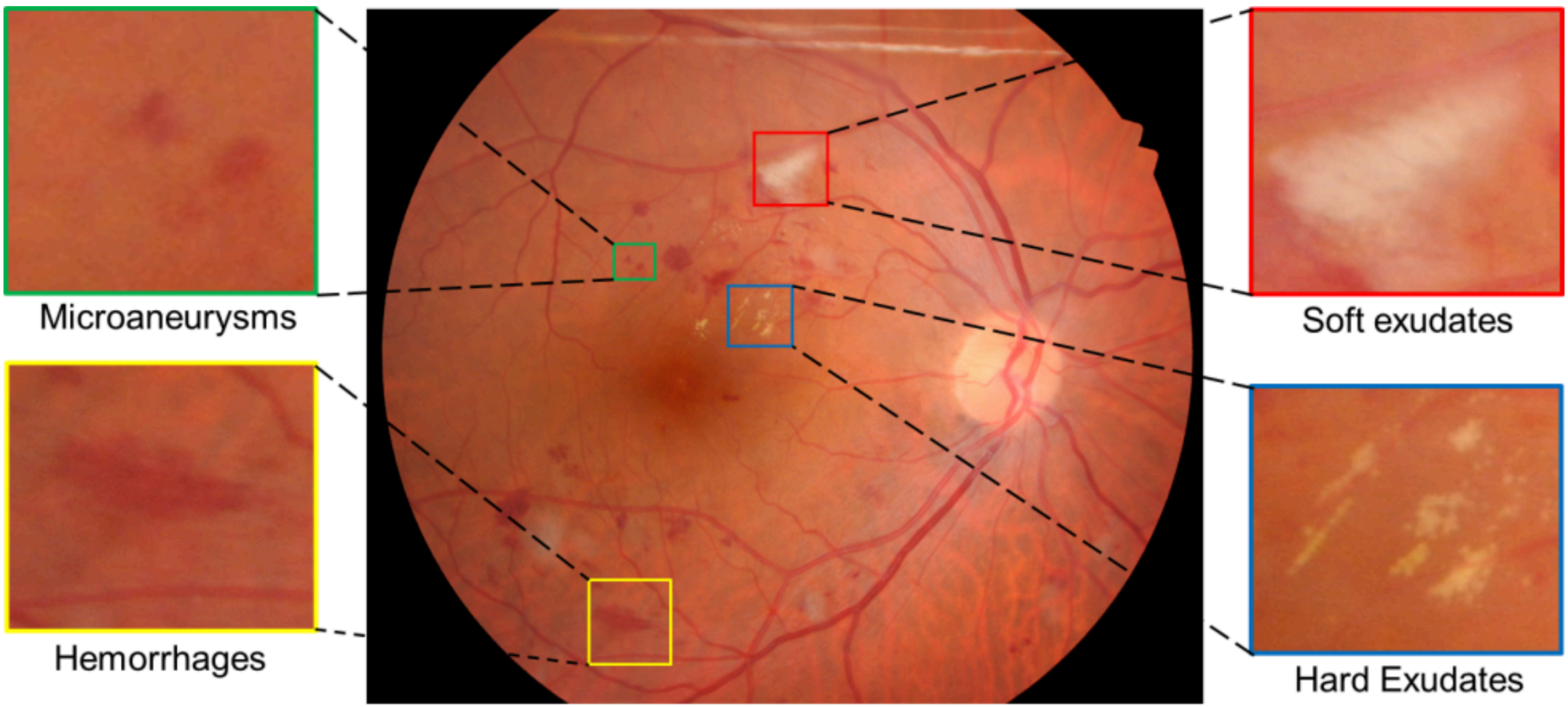
CONTENTS

EXPERIMENTS AND RESULTS

Dataset

- *Indian Diabetic Retinopathy Image Dataset* (IDRiD) from the *Diabetic Retinopathy: Segmentation and Grading Challenge*
- 54 images with different retinal lesions:
    - Microaneurysms (MA)
    - Haemorrhages (HE)
    - Soft Exudates (SE)
    - Hard Exudates (EX)
- Dataset is divided in: 80 % train, 10% validation and 10% test
- Images are divided in patches to fit in memory

EXPERIMENTS AND RESULTS



Retinal lesions in dataset images

EXPERIMENTS AND RESULTS

U-Net

- 19 convolutional layers
- Batch normalization
- Learning rate: 0.0001
- Batch size: 8

Optimizer	DA	Loss function	EX	HE	MA	SE
Adam	False	Categorical cross-entropy	0.579	0.365	0.374	0.281
Adam	False	Dice coefficient	**0.727**	0.429	0.423	0.0
Adam	True	Generalised dice coefficient	0.694	**0.458**	**0.477**	**0.351**

Results using U-Net with dice score metric

EXPERIMENTS AND RESULTS

U-Net

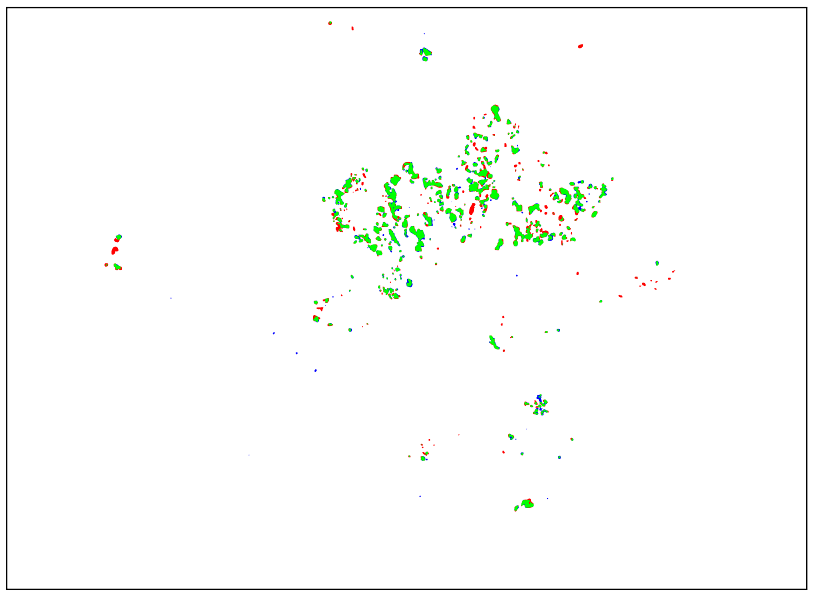
- 19 convolutional layers
- Batch normalization
- Learning rate: 0.0001
- Batch size: 8

Optimizer	DA	Loss function	EX	HE	MA	SE
Adam	False	Categorical cross-entropy	0.579	0.365	0.374	0.281
Adam	False	Dice coefficient	**0.727**	0.429	0.423	0.0
Adam	True	Generalised dice coefficient	0.694	**0.458**	**0.477**	**0.351**

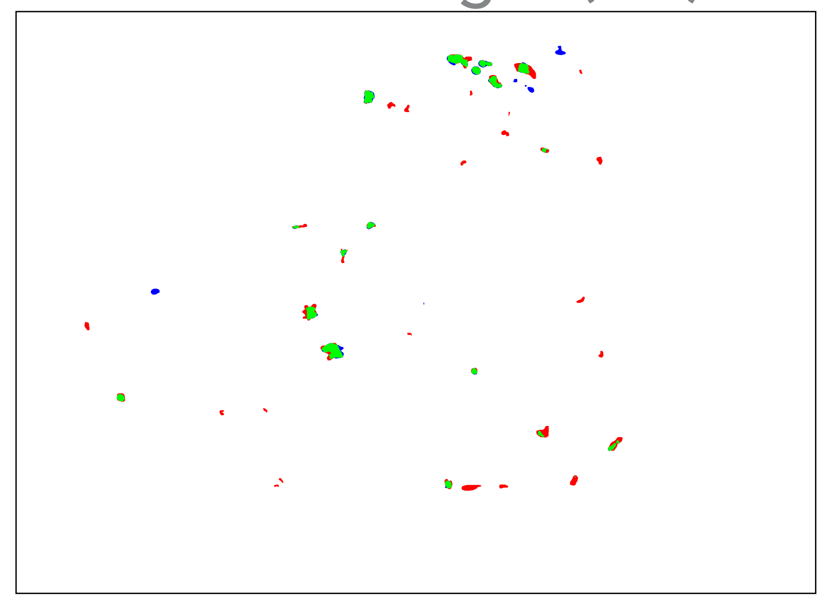
Results using U-Net with dice score metric

EXPERIMENTS AND RESULTS **U-Net**

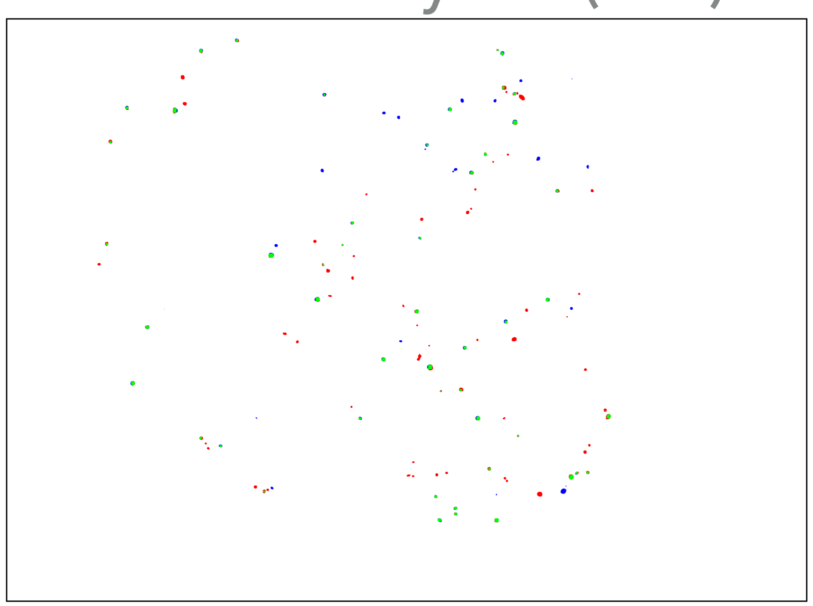
Hard Exudates (EX)



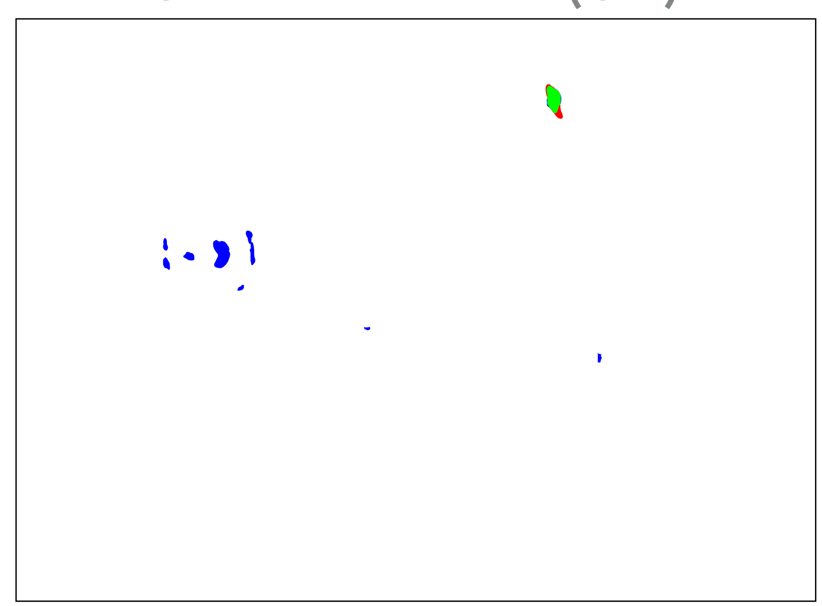
Haemorrhages (HE)



Microaneurysms (MA)



Soft Exudates (SE)



Difference maps. Green: true positives, Red: false negatives, Blue: false positives

EXPERIMENTS AND RESULTS

Adversarial training

$$L_{GAN} = \lambda \cdot L_{ADV} + L_{SEG}$$

$$L_{SEG} = \text{Generalised dice coefficient}$$

L_{ADV}	λ	EX	HE	MA	SE
Binary cross-entropy	0.05	**0.734**	0.458	0.423	0.415
Binary cross-entropy	0.07	0.713	0.439	0.454	**0.470**
Mean Squared Error	0.05	0.707	**0.476**	**0.462**	0.405
Mean Squared Error	0.07	0.722	0.472	0.447	0.364

Results using U-Net with adversarial training with dice score metric

EXPERIMENTS AND RESULTS

Adversarial training

$$L_{GAN} = \lambda \cdot L_{ADV} + L_{SEG}$$

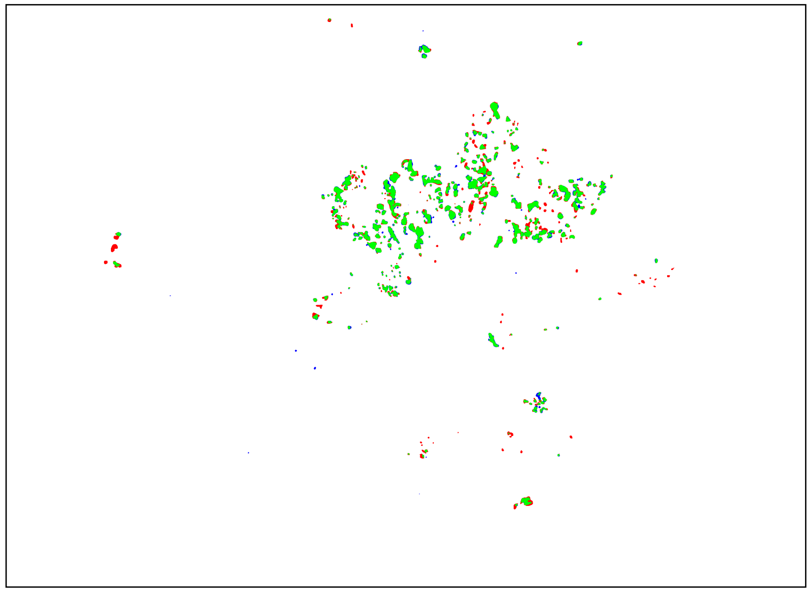
$$L_{SEG} = \text{Generalised dice coefficient}$$

L_{ADV}	λ	EX	HE	MA	SE
Binary cross-entropy	0.05	**0.734**	0.458	0.423	0.415
Binary cross-entropy	0.07	0.713	0.439	0.454	**0.470**
Mean Squared Error	0.05	0.707	**0.476**	**0.462**	0.405
Mean Squared Error	0.07	0.722	0.472	0.447	0.364

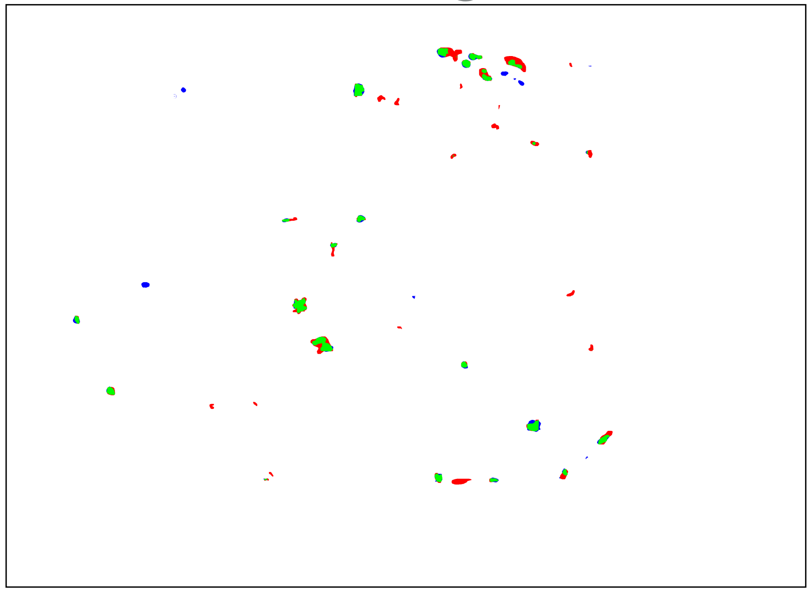
Results using U-Net with adversarial training with dice score metric

EXPERIMENTS AND RESULTS **Adversarial training**

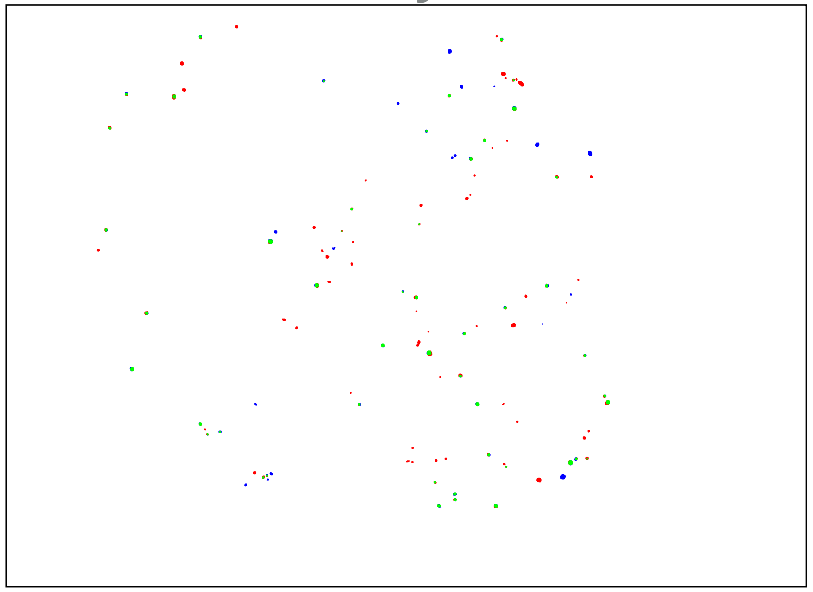
Hard Exudates (EX)



Haemorrhages (HE)



Microaneurysms (MA)



Soft Exudates (SE)

Difference maps. Green: true positives, Red: false negatives, Blue: false positives

EXPERIMENTS AND RESULTS

Discussion

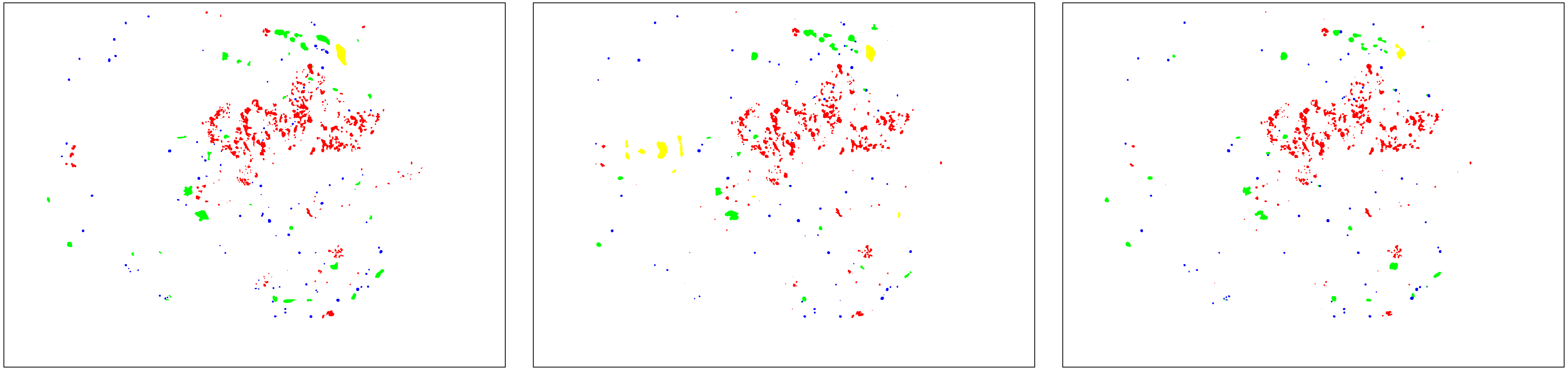
- Criteria to rank the results: we focused on class HE
- The addition of adversarial training improve the results

Architecture	EX	HE	MA	SE
U-Net without adversarial training	0.694	0.458	0.477	0.351
U-Net with adversarial training	0.707	0.476	0.462	0.405

Comparison of the addition of adversarial training using dice score metric

EXPERIMENTS AND RESULTS

Ground truth | Predicted labels without adv. training | Predicted labels with adv. training



Red: class Hard Exudates (EX), Blue: class Microaneurysms (MA),
Green: class Haemorrhages (HE), Yellow: class Soft Exudates (SE)

CONTENTS

CONCLUSIONS AND FUTURE DEVELOPMENT

- We proved that adversarial training applied to a segmentation network improved its performance
- The architecture's configuration must take into account the high-imbalance between classes
- Results are not comparable to the ones from the challenge due to the metric used

CONCLUSIONS AND FUTURE DEVELOPMENT

- As future development, a more robust estimate of the model's performance could be done
- The use of an architecture with dilated convolutions instead of U-Net to avoid pooling operations
- The use of an adaptive sampling scheme would also be interesting to tackle the high-imbalance between classes

THANK YOU FOR YOUR ATTENTION

NATÀLIA GULLÓN

July 2018