

Towards landmarks prediction with Deep Network

Van-Linh LE^{1,3}, Marie BEURTON-AIMAR¹,
Akka ZEMMARI¹, Nicolas PARISEY²

linhlv@dlu.edu.vn, van-linh.le@labri.fr, beurton@labri.fr
akka.zemmari@labri.fr, nicolas.parisey@inra.fr

¹LaBRI-CNRS 5800, Bordeaux University, France

²IGEPP, INRA 1349 Rennes, France

³ITDLU, Dalat University, Vietnam

ICPRS Conference

Valparaíso, 22-24 May, 2018

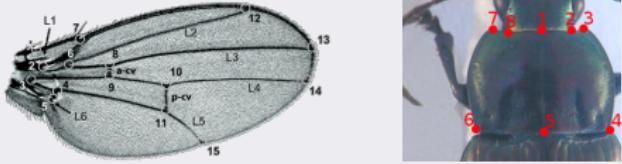


Morphometry analysis

- ▶ Used to study the complex interaction between the morphometry of species and environmental factors.
- ▶ Characterize information of biological species such as shape, sizes, or **landmarks**, . . .

Landmark

- ▶ A kind of **point of interest**
- ▶ A specific point defined by biologist. For example, intersection of veins on fly wing, the corner of beetle's pronotum shape, . . .



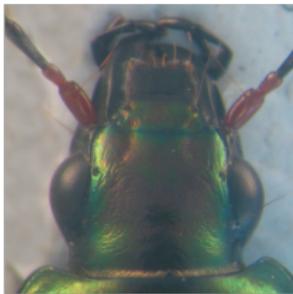
Dataset



- ▶ Images have been taken from 293 **beetles**, separate into 5 parts
- ▶ Format: 2D in RGB color
- ▶ Focus on **pronotum** images



Body part/elytre



Head part/tete



Pronotum part



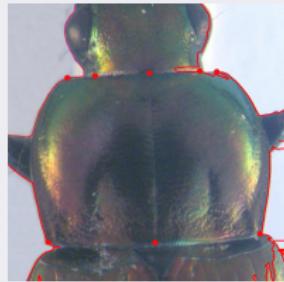
Manual landmarks

- ▶ Time-consuming
- ▶ Difficult to reproduce

Problems

Pronotum image:

- ▶ Not precised: contains also a part of head and body
- ▶ Difficult to segment this object
- ▶ The landmarks are set both on the shape and inside the object





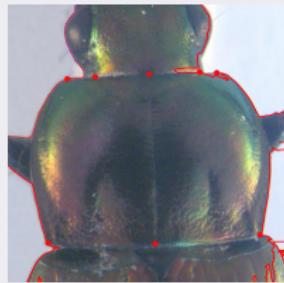
Manual landmarks

- ▶ Time-consuming
- ▶ Difficult to reproduce

Problems

Pronotum image:

- ▶ Not precised: contains also a part of head and body
- ▶ Difficult to segment this object
- ▶ The landmarks are set both on the shape and inside the object



How to automatically predict the landmarks coordinates?

Content



Deep learning and Convolutional Neural Networks

Deep learning

Convolutional neural networks (CNNs)

Proposed method

Network architectures

Data augmentation

Results

Training from scratch

Fine-tuning

Conclusion



Definition¹

- ▶ A class of machine learning methods.
- ▶ Use a cascade of multiple layers for feature extraction and transformation.
- ▶ Learn multiple levels of representation in supervised or unsupervised mode.

¹ Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015



Definition¹

- ▶ A class of machine learning methods.
- ▶ Use a cascade of multiple layers for feature extraction and transformation.
- ▶ Learn multiple levels of representation in supervised or unsupervised mode.

Applications

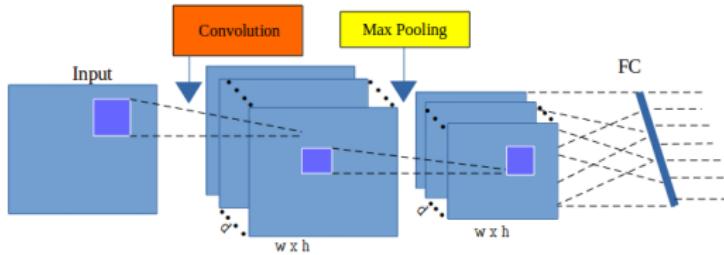
- ▶ Computer vision (image recognition and classification)
- ▶ Speech recognition
- ▶ Question answering, language translation, ...

¹ Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015

Convolutional neural networks



- ▶ Consists an input, an output and multiple hidden layers¹
- ▶ Arranges the data in 3 dimensions: *width, height and depth*
- ▶ Classical layers: **convolutional** layers, **pooling** layers, **dropout** layers, **full-connected** layers, ...



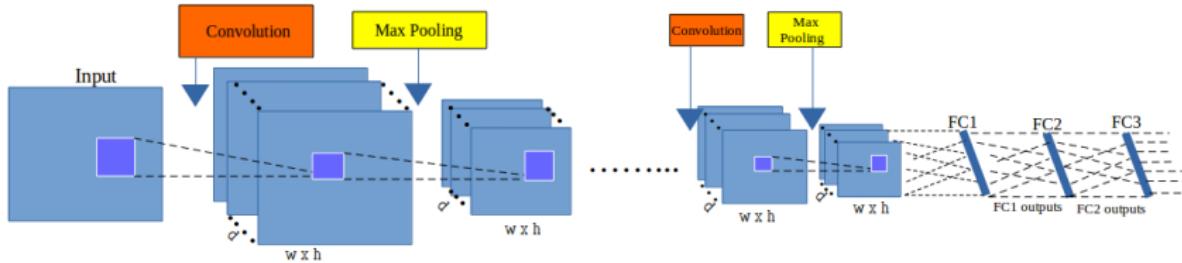
¹ Y. LeCun et al, "Convolutional networks and applications in vision", 2010.

First architecture



The first applied networks:

- ▶ Repeat the pair of **convolutional** and **maximum pooling** layers
- ▶ Trying to adjust the parameters of the layers



Results are not satisfying

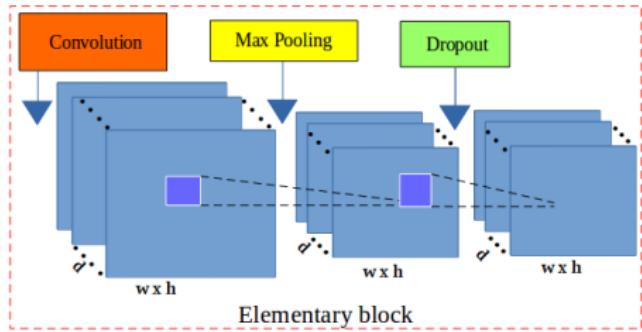
Our proposed architecture

Elementary block



Elementary block:

- ▶ A **convolutional** layer
- ▶ A **maximum pooling** layer
- ▶ A **dropout** layer



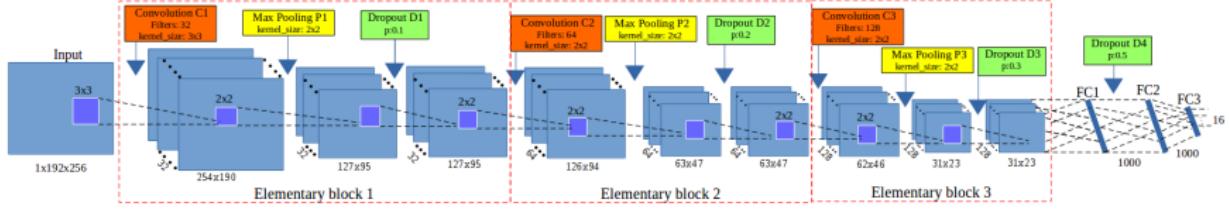
Our proposed architecture

Elementary blocks composition



The proposed model:

- ▶ Three elementary blocks
- ▶ Three full-connected (FC) layers
- ▶ A dropout layer was inserted between the first and the second FC layer



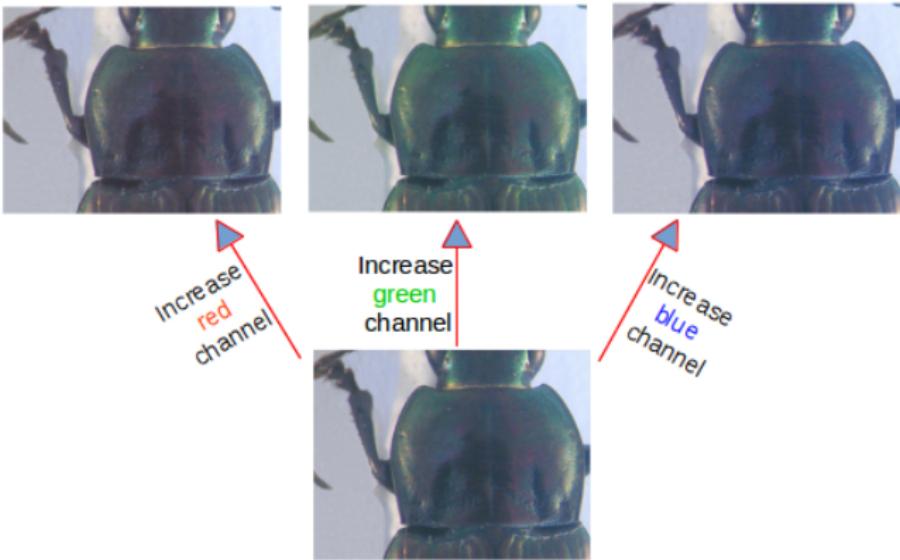
Data augmentation



Dataset: **293 pronotum** images in **RGB** format.

Augmentation methods:

- ▶ Increase the value of each channel



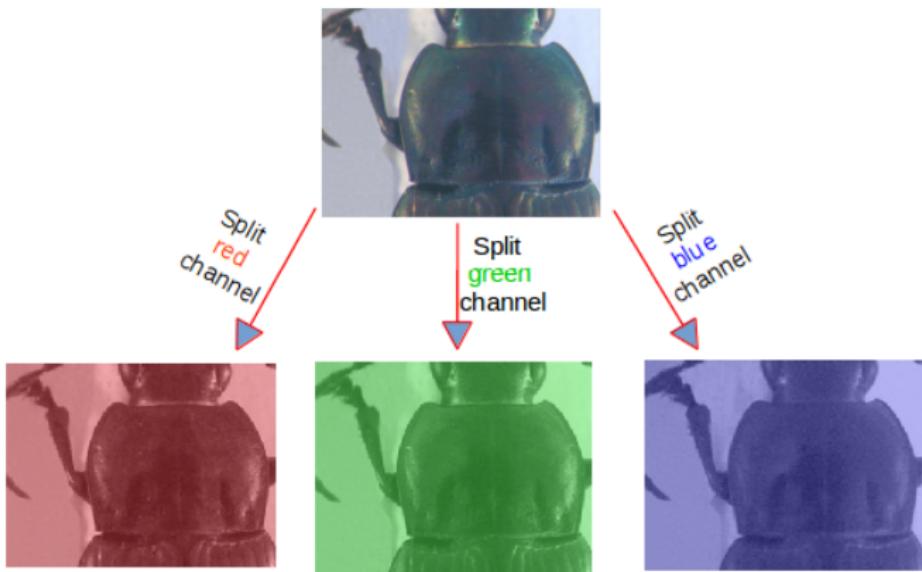
Data augmentation



Dataset: 293 pronotum images in **RGB** format.

Augmentation methods:

- ▶ Split the channels



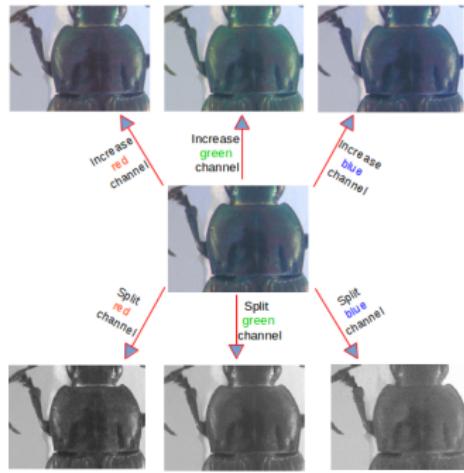
Data augmentation



Dataset: 293 pronotum images in **RGB** format.

Augmentation methods:

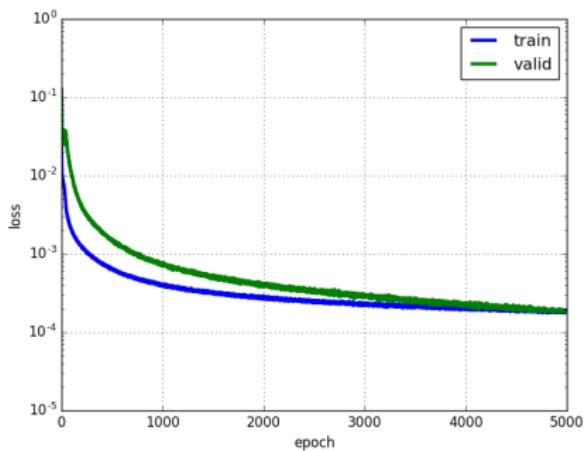
- ▶ Increase the value of each channel
 - ▶ Split the channels
- ⇒ Total: $293 \times 7 = 2,051$ images



Training



- ▶ Training dataset: 1,820 images (260×7)
- ▶ Apply the cross-validation to select training data
- ▶ Training parameters: momentum ($0.9 \rightarrow 0.9999$), learning rate ($0.03 \rightarrow 0.00001$), 5000 epochs¹
- ▶ Image shows training and validation losses of the model
- ▶ Training time: 3 hours using NVIDIA TITAN X card



1.

V.L. Le, M. Beurton-Aimar, A. Zemmari, N. Parisey - Using the full training set.

ICPRs-18 Conference

First results

Correlation metrics and landmarks on the images



- ▶ Quality metrics: coefficient of determination (r^2), explained variance (EV), Pearson correlation.

Metric	r^2	EV	Pearson
Proposed architecture	0.9952	0.9951	0.9974

First results

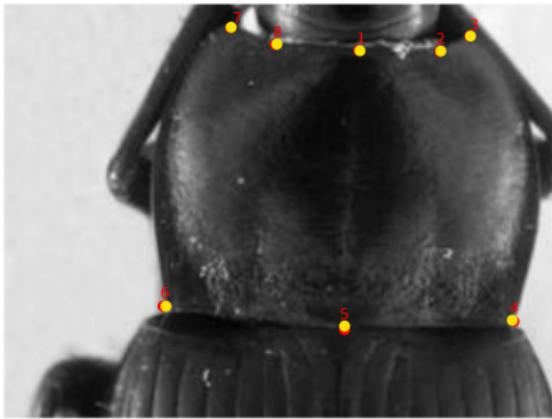
Correlation metrics and landmarks on the images



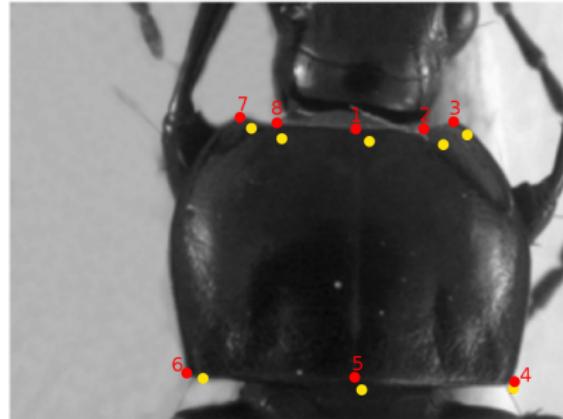
- ▶ Quality metrics: coefficient of determination (r^2), explained variance (EV), Pearson correlation.

Metric	r^2	EV	Pearson
Proposed architecture	0.9952	0.9951	0.9974

- ▶ Display the landmarks on the images:



(a)



(b)

First results

Average distances



- ▶ Calculate the distance between predicted landmarks and corresponding manual landmarks.
- ▶ Compute the average distance by landmark.

Landmark	Distance (in pixels)
1	4.002
2	4.4831
3	4.2959
4	4.3865
5	4.2925
6	5.3631
7	4.636
8	4.9363

The statistic of average distances on all images per landmark.

Transfer learning/Knowledge transfer



- ▶ Re-use model developed for a specific task/dataset to lead another task on another dataset
- ▶ **Fine-tuning:** retrain a pretrained model
- ▶ **Model Zoo** (Caffe library): people share their pre-trained networks.

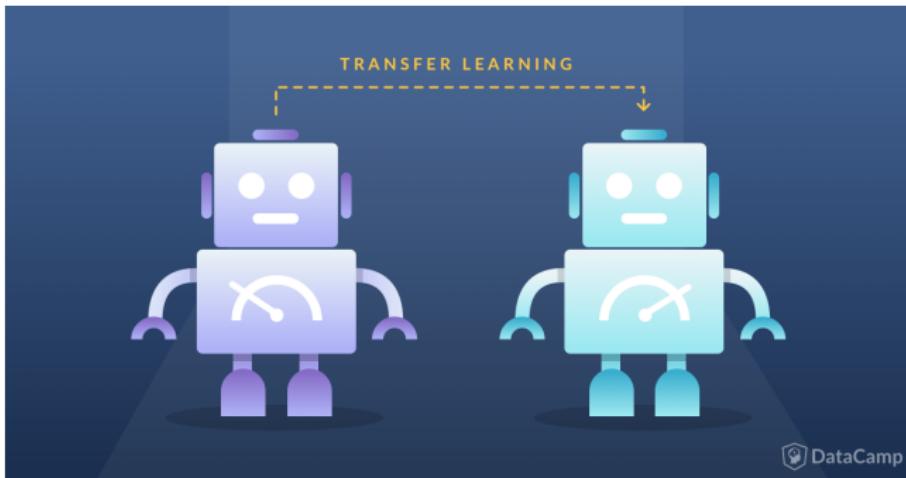
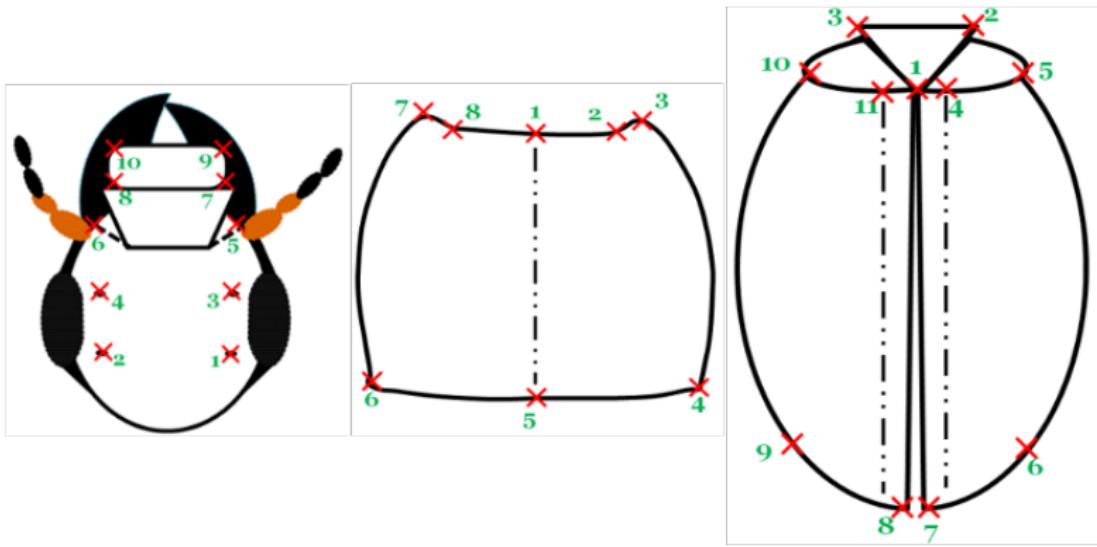


Image source: DataCamp

Fine-tuning our model



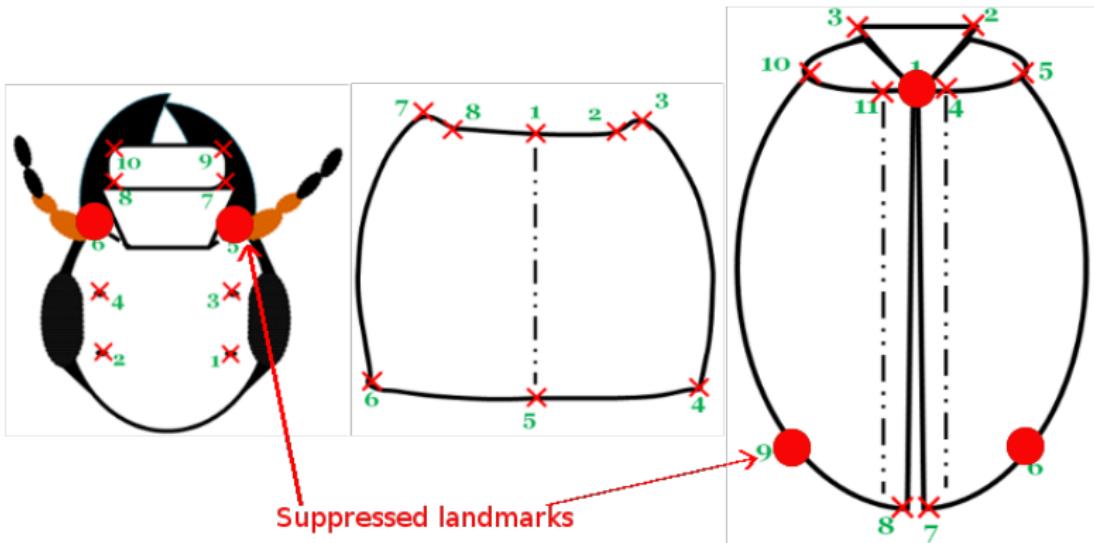
- ▶ Estimated landmarks on pronotum images when fine-tuning on **VGG-16, VGG-19, ResNet50** have not been improved
- ▶ Train the model on a dataset including the images of 3 parts of beetles: head, body and pronotum parts (**5,460 images**)
- ▶ Fine-tune pretrained model on pronotum dataset



Fine-tuning our model



- ▶ Estimated landmarks on pronotum images when fine-tuning on **VGG-16, VGG-19, ResNet50** have not been improved
- ▶ Train the model on a dataset including the images of 3 parts of beetles: head, body and pronotum parts (**5,460 images**)
- ▶ Fine-tune pretrained model on pronotum dataset



Results

A comparation of average distances



Comparing the average distances between two processes: training from scratch and fine-tuning.

Landmarks	From scratch		With fine-tuning	
	Average	SD	Average	SD
LM1	4.002	2.5732	2.486	1.5448
LM2	4.4831	2.7583	2.7198	1.7822
LM3	4.2959	2.7067	2.6523	1.8386
LM4	4.3865	3.0563	2.7709	1.9483
LM5	4.2925	2.9086	2.4872	1.6235
LM6	5.3631	3.4234	3.0492	1.991
LM7	4.636	2.8426	2.6836	1.7781
LM8	4.9363	3.0801	2.8709	1.9662

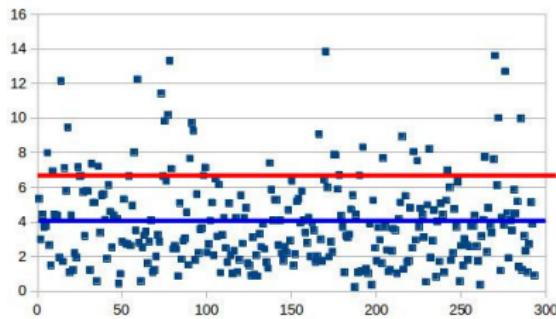
Results

Distribution of average distances

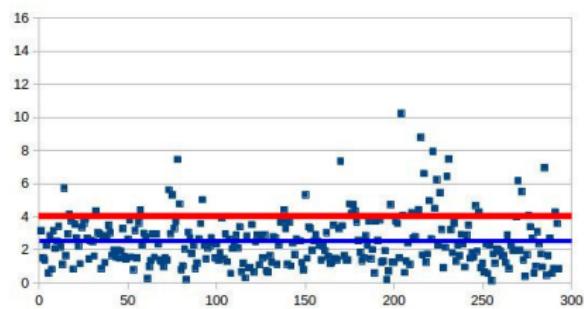


17

- The distribution of distance of the best result: 1st landmark



(a) Training from scratch



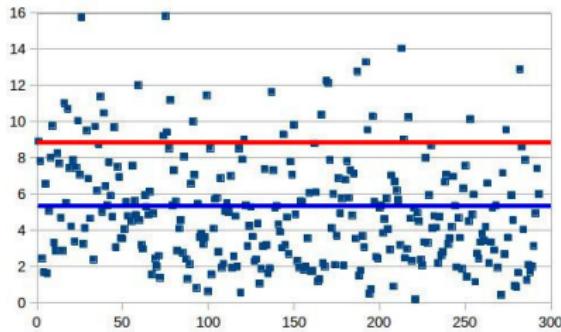
(b) With fine-tuning

Results

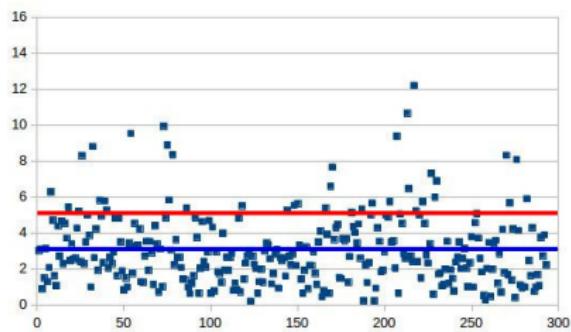
Distribution of average distances



- The distribution of distance of the worst result: 6th landmark



(a) Training from scratch



(b) With fine-tuning



Conclusion

- ▶ Propose a new CNN architecture with elementary blocks to predict the landmarks on pronotum images.
- ▶ Propose a new procedure to augment the dataset.
- ▶ Apply fine-tuning to improve the quality of predicted landmarks.
- The predicted landmarks able to replace the manual landmarks without segmentation step.

Conclusion



Conclusion

- ▶ Propose a new CNN architecture with elementary blocks to predict the landmarks on pronotum images.
- ▶ Propose a new procedure to augment the dataset.
- ▶ Apply fine-tuning to improve the quality of predicted landmarks.
- The predicted landmarks able to replace the manual landmarks without segmentation step.

Future works

- ▶ Applying the method on body and head parts
- ▶ Going deeply how to design the right pre-training model



Thank you for attention!