Unsupervised Segmentation of Music Symbols using a GAN Detector

Kwon-Young Choi [1]
Supervisors : Bertrand Coüasnon [1] Yann Ricquebourg [1]
Richard Zanibbi [2]

[1]Intuidoc, Université de Rennes, CNRS, IRISA, F-35000, France
[2]DPRL, Rochester Institute of Technology, New-York, USA

kwon-young.choi@irisa.fr

2019-06-11

Plan

1 Background

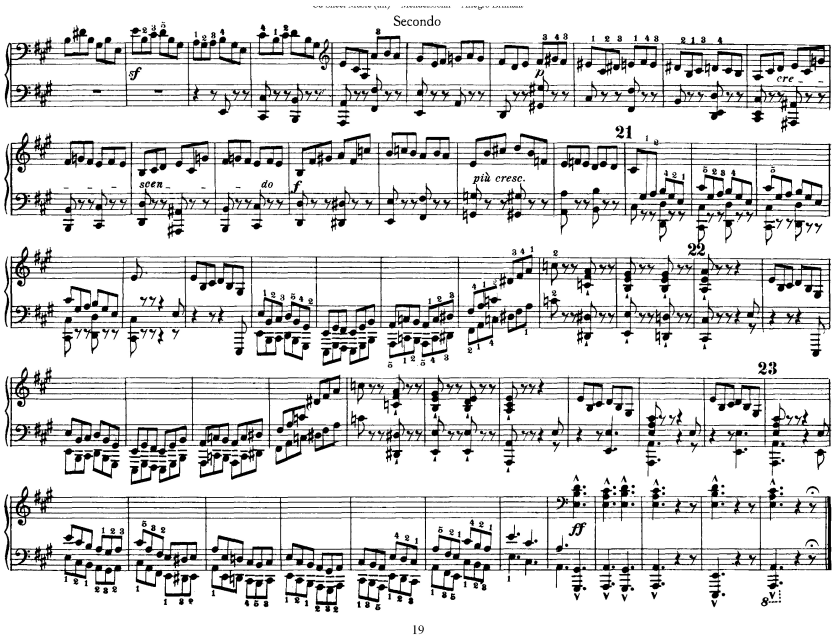
2 Accidental Detection

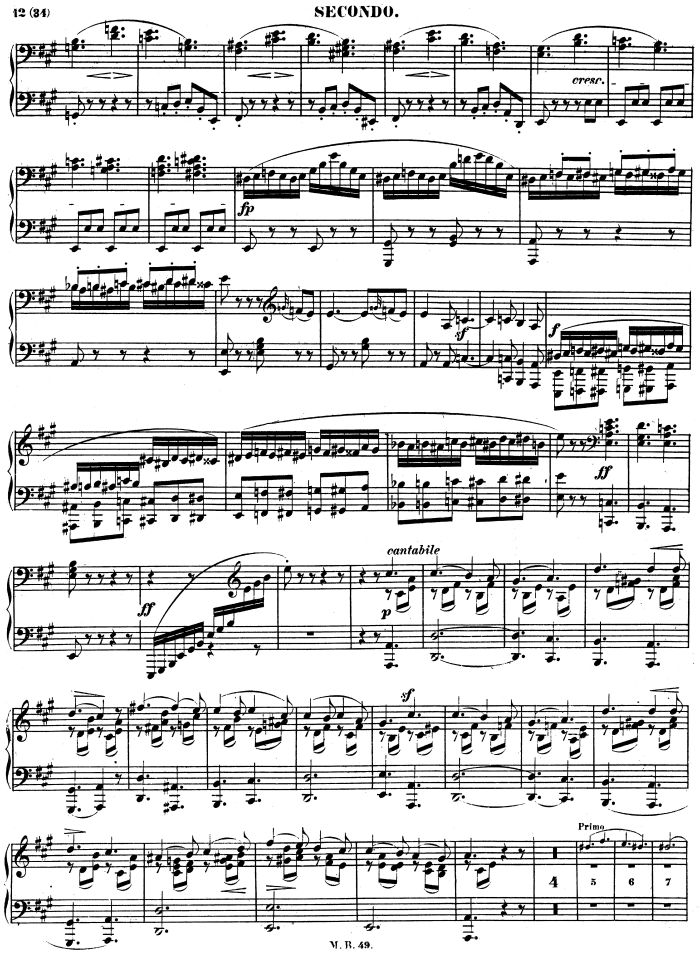
3 SegGAN

4 Conclusion

Optical Music Recognition

1. Complex, dense, noisy orchestral scores





Optical Music Recognition

- Complex, dense, noisy orchestral scores
- Segmentation problems

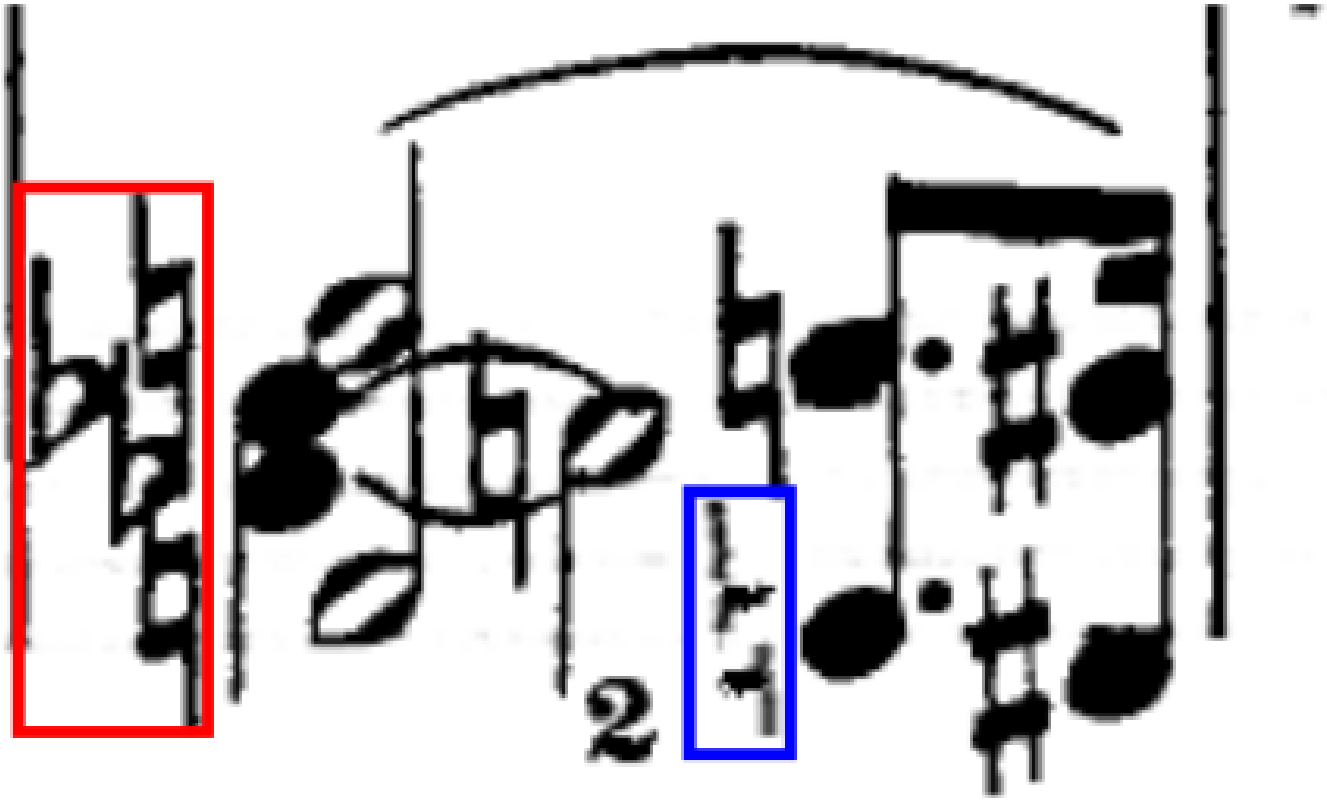
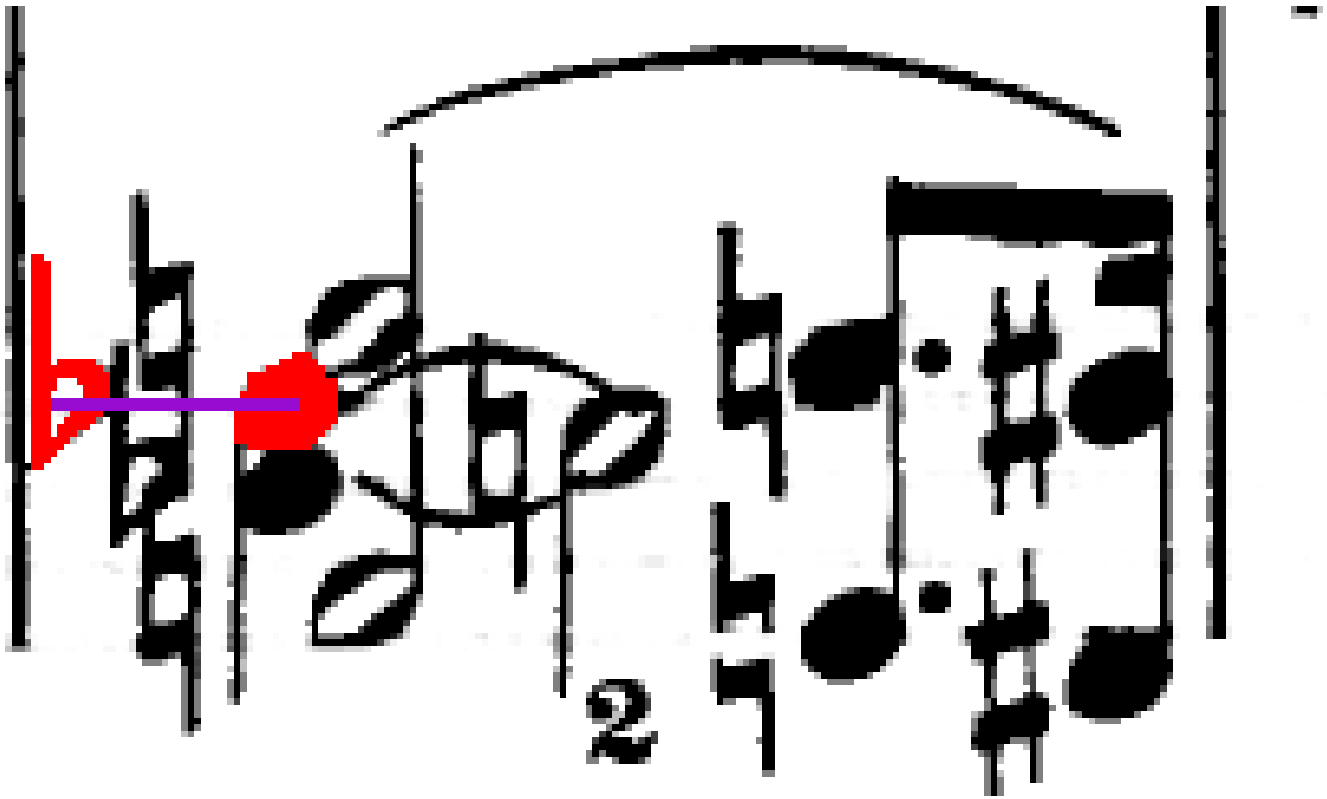


Figure: Touching symbols, Broken symbols

Optical Music Recognition

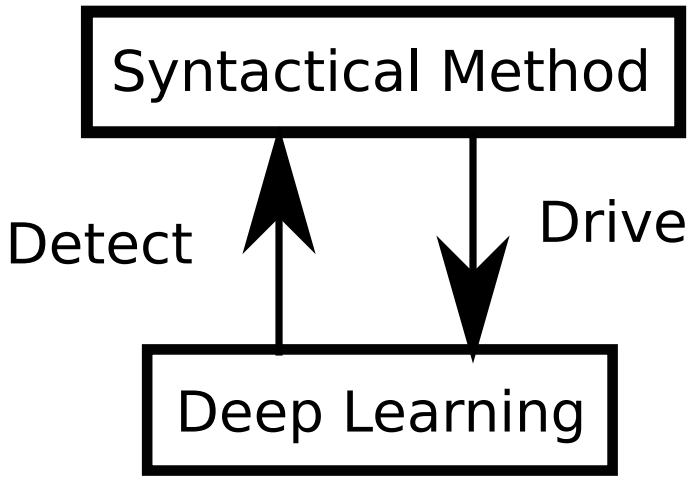
- Complex document and notation
- Segmentation problems
  - Importance of good localization!



PhD Focus

Context

- Syntactical recognition for modeling the musical notation and driving deep learning
- Solve segmentation problems with Deep learning for detecting/recognizing symbols



PhD Focus

Objectives

- How to detect/recognize music symbols?
- How to reduce the quantity of annotated data?
  - Syntactical analysis allows generation of subimage with contextual information
  - Bootstrapping samples
  - Synthetic data

Goal

- Build OMR system using a unified set of rule for music notation
- Adapt system to the data using Deep Learning
  - Without requiring heavy manual annotations

Plan

1 Background

2 Accidental Detection

3 SegGAN

4 Conclusion

Single Accidental Detection + Rejection

- Complex music notation ⇒ Rule based OMR system
  - Contextual knowledge

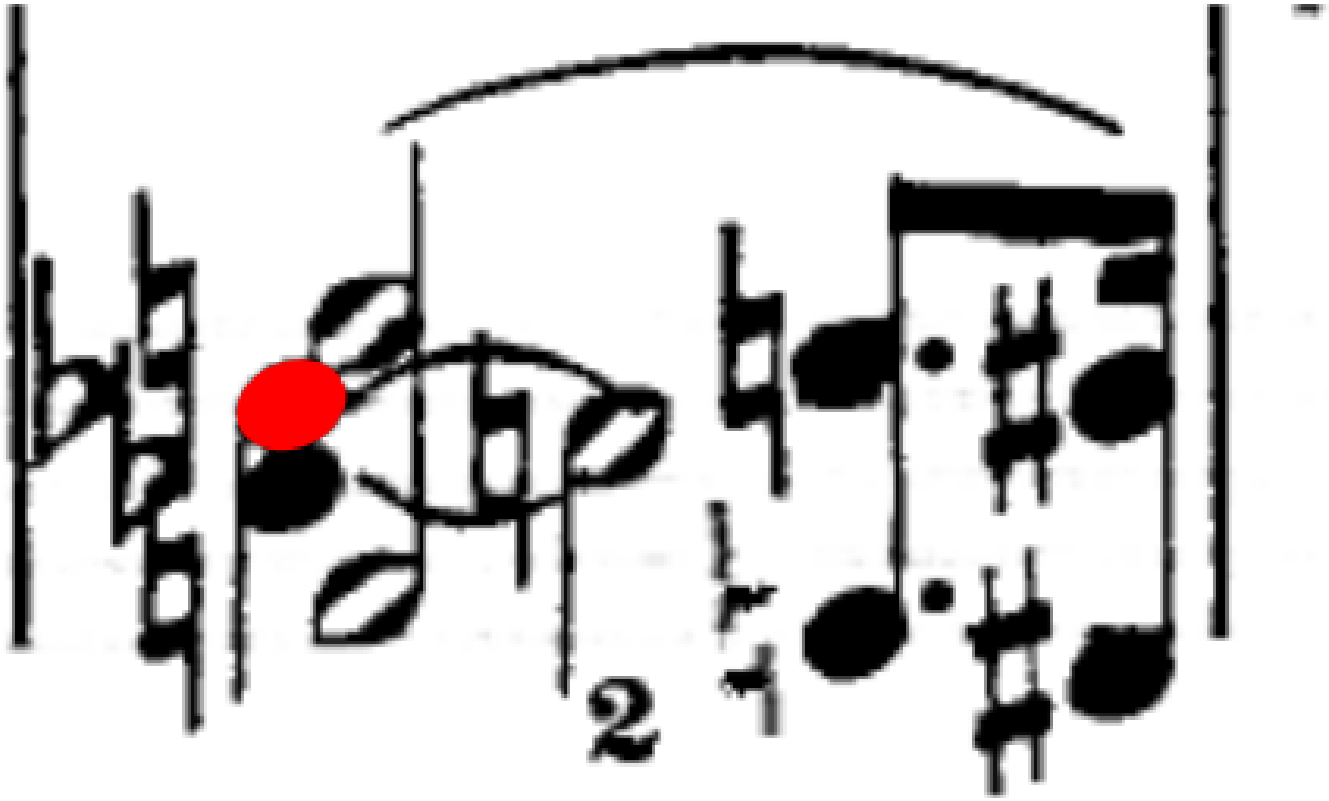


Figure: Note head position known

Single Accidental Detection + Rejection

- Complex music notation ⇒ Rule based OMR system
  - Contextual knowledge
- Segmentation problems ⇒ Deep Learning detectors
  - Produce localization + class information

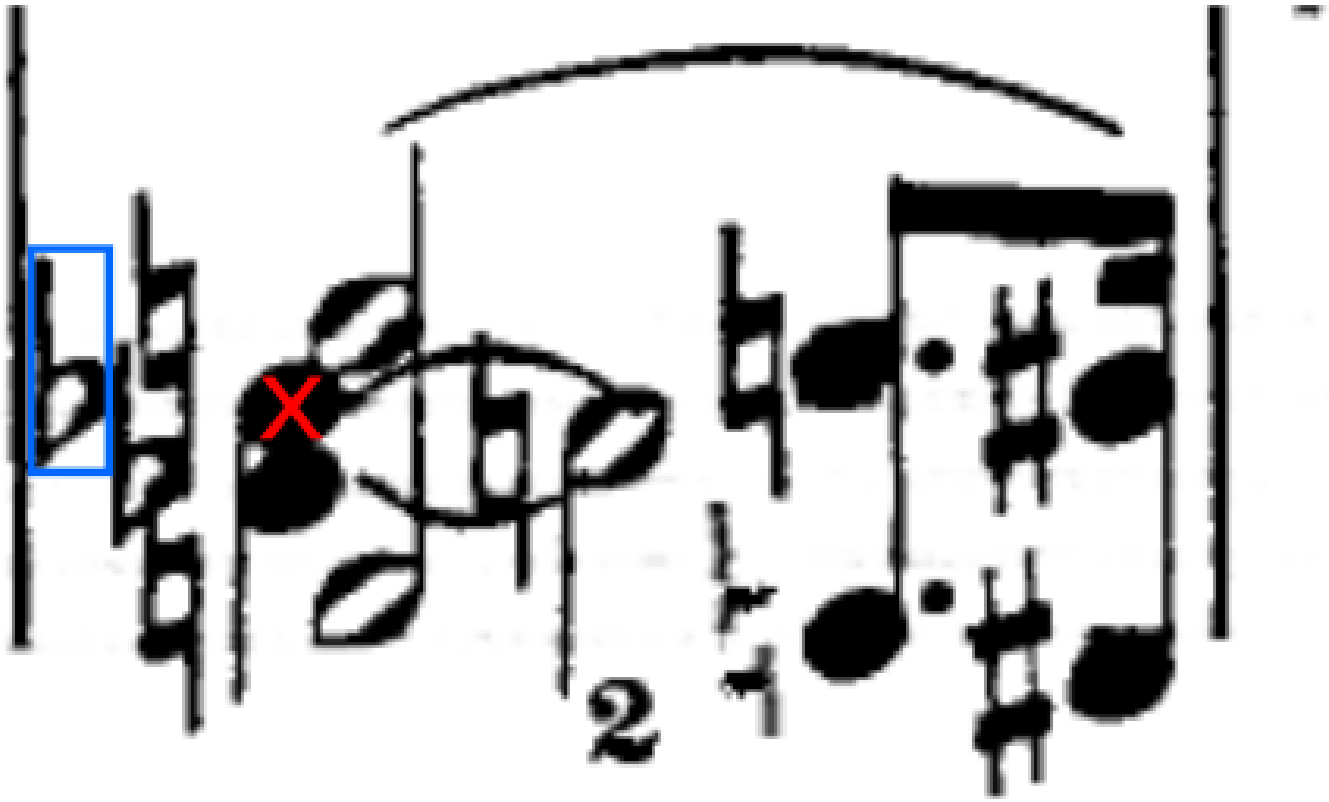


Figure: Accidental detection

Single Accidental Detection + Rejection

- Complex music notation ⇒ Rule based OMR system
  - Contextual knowledge
- Segmentation problems ⇒ Deep Learning detectors
  - Produce localization + class information

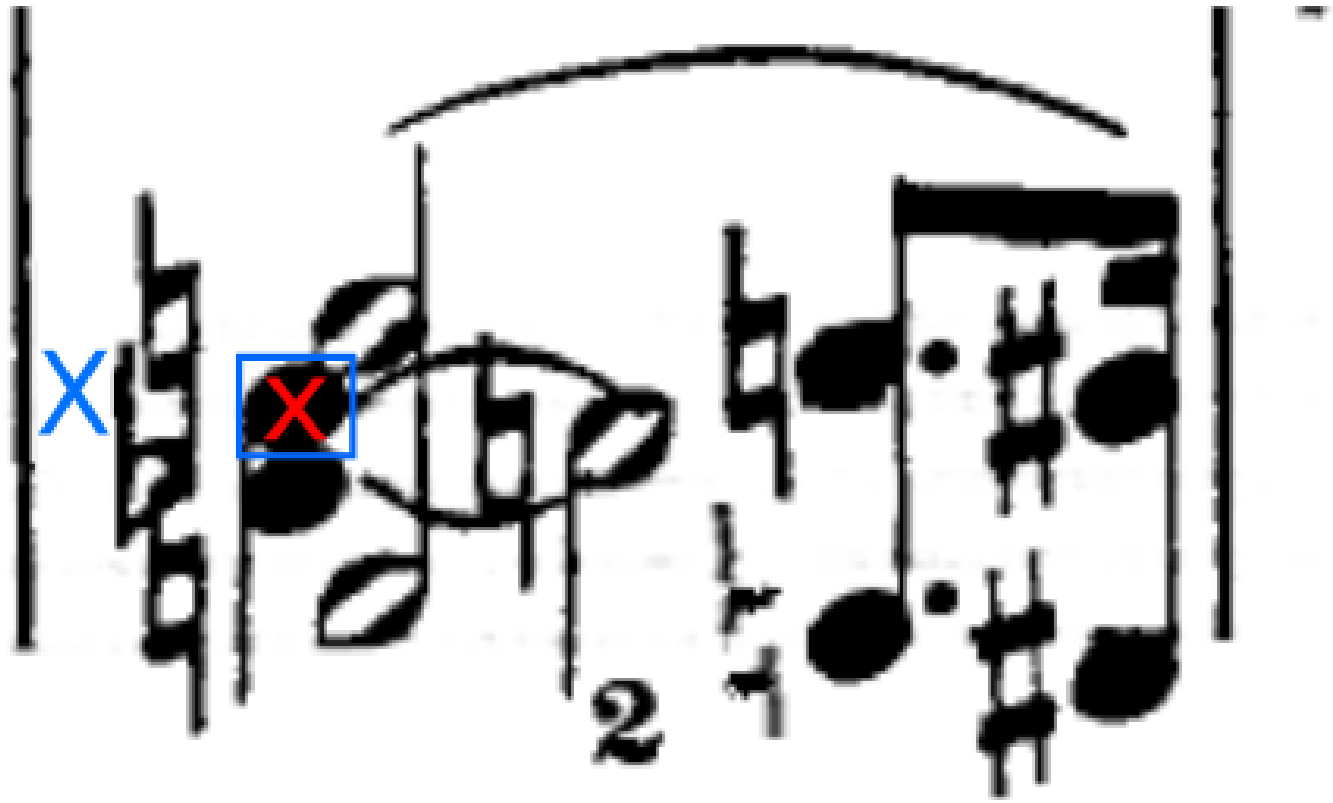


Figure: Rejection detection

Single Accidental Detection + Rejection

- Complex music notation ⇒ Rule based OMR system
  - Contextual knowledge
- Segmentation problems ⇒ Deep Learning detectors
  - Produce localization + class information
- Contextual information: limit the search space

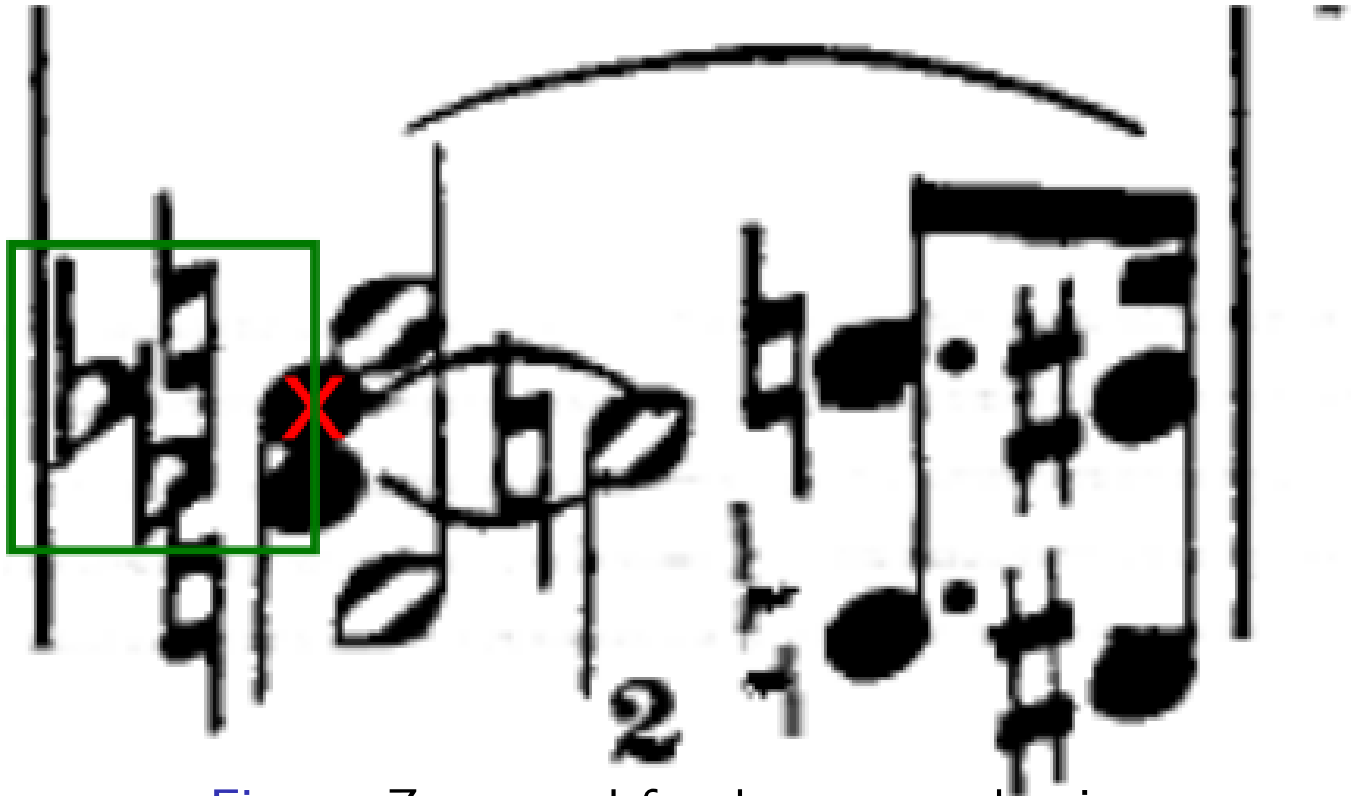


Figure: Zone used for dataset production

Single Accidental Detection + Rejection

Figure: Dataset examples. In absence of accidental, it's considered as Rejection

Dataset Constitution

Table: Dataset composition

Label	**Quantity**
No accidental (Reject)	968
Natural	968
Sharp	777
Flat	242
Total	**2955**

- Square image size: 4x size of interline, resized to 120x120 pixels
  - Interline: Vertical distance between two staff lines + height of one staff line
- 2955 samples $\Rightarrow$ Small dataset

Bootstrapping

⇒ Generate **more data** for localization task.

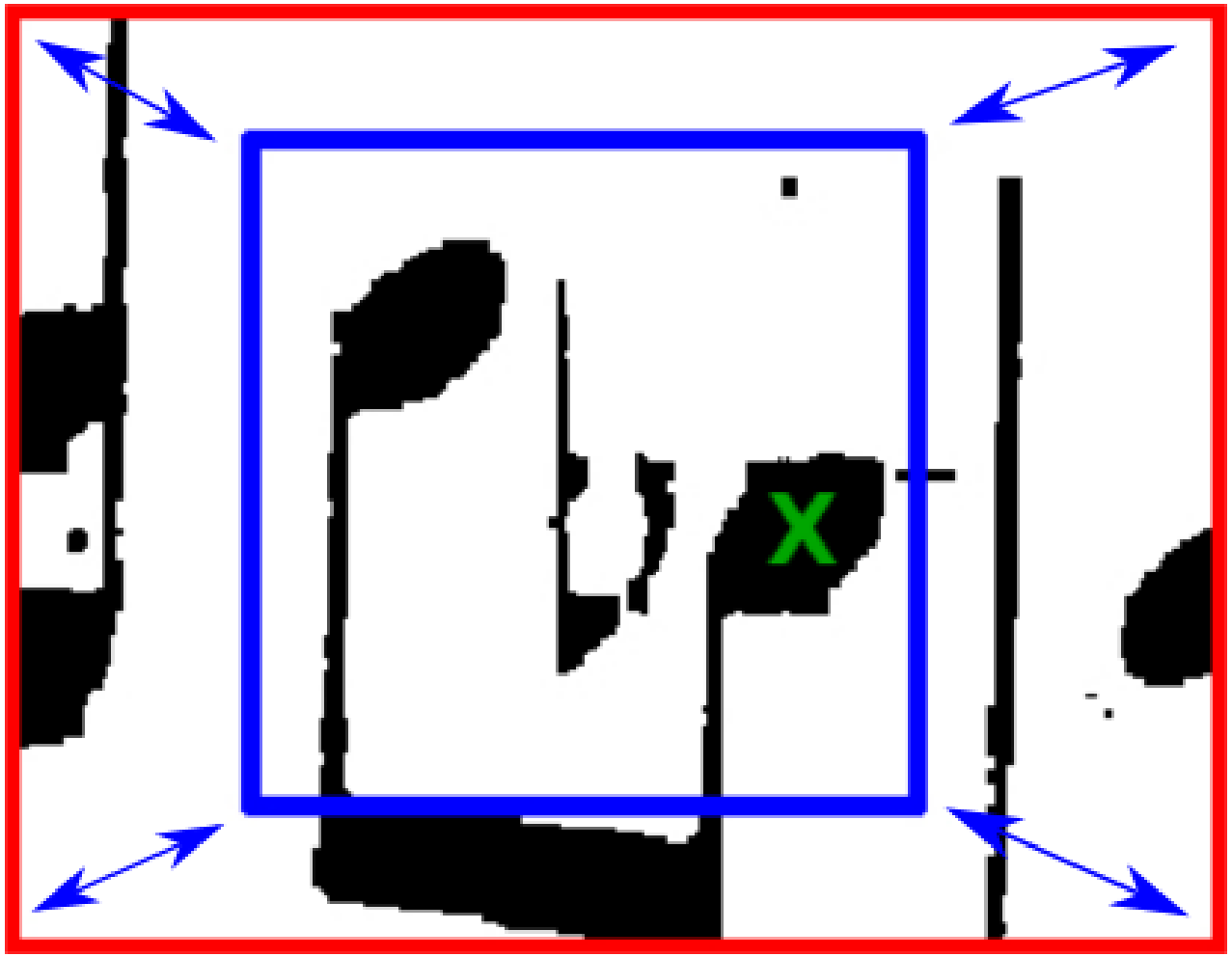


Figure: Random sampling on different zones

Bootstrapping

⇒ Generate **more data** for localization task.

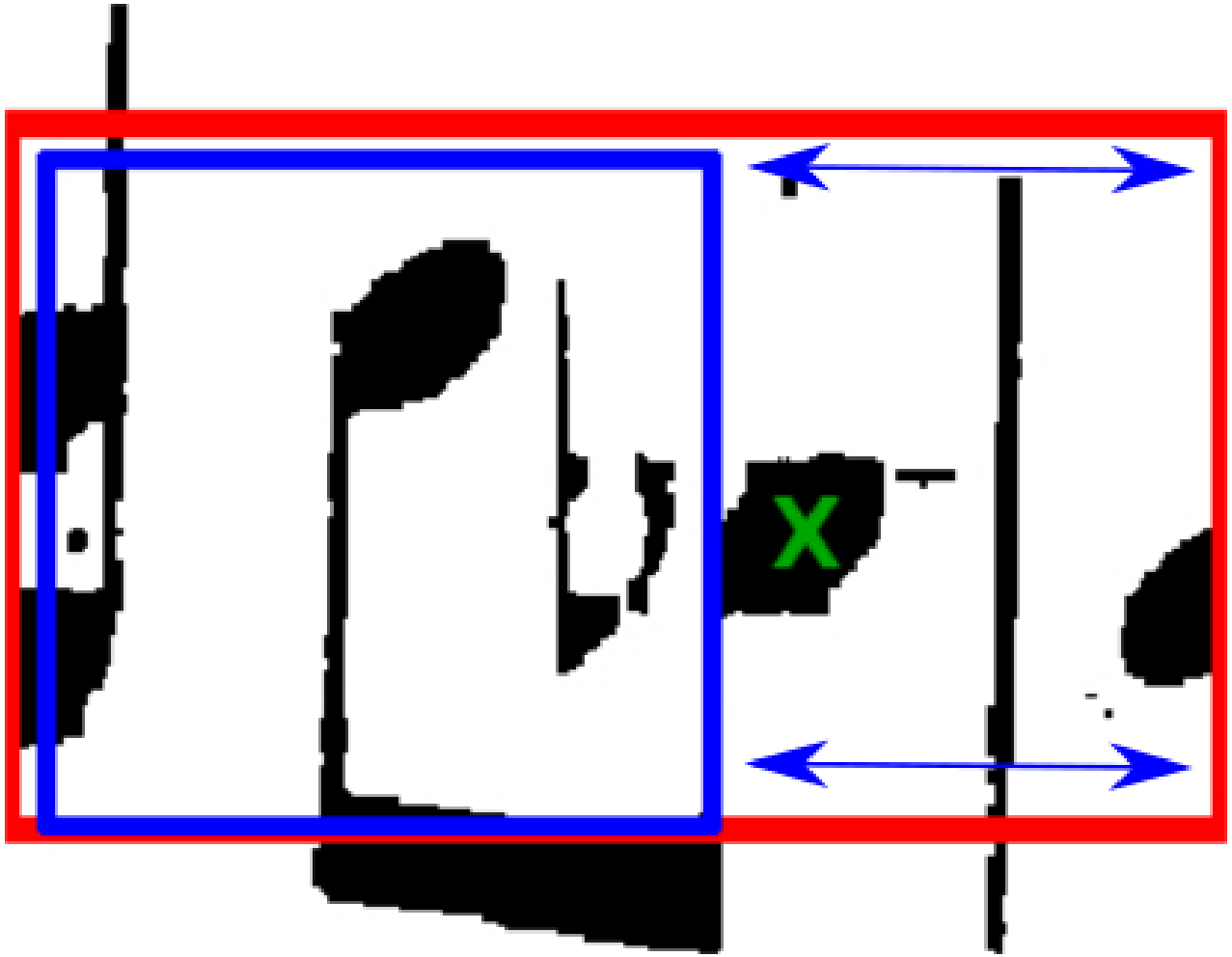
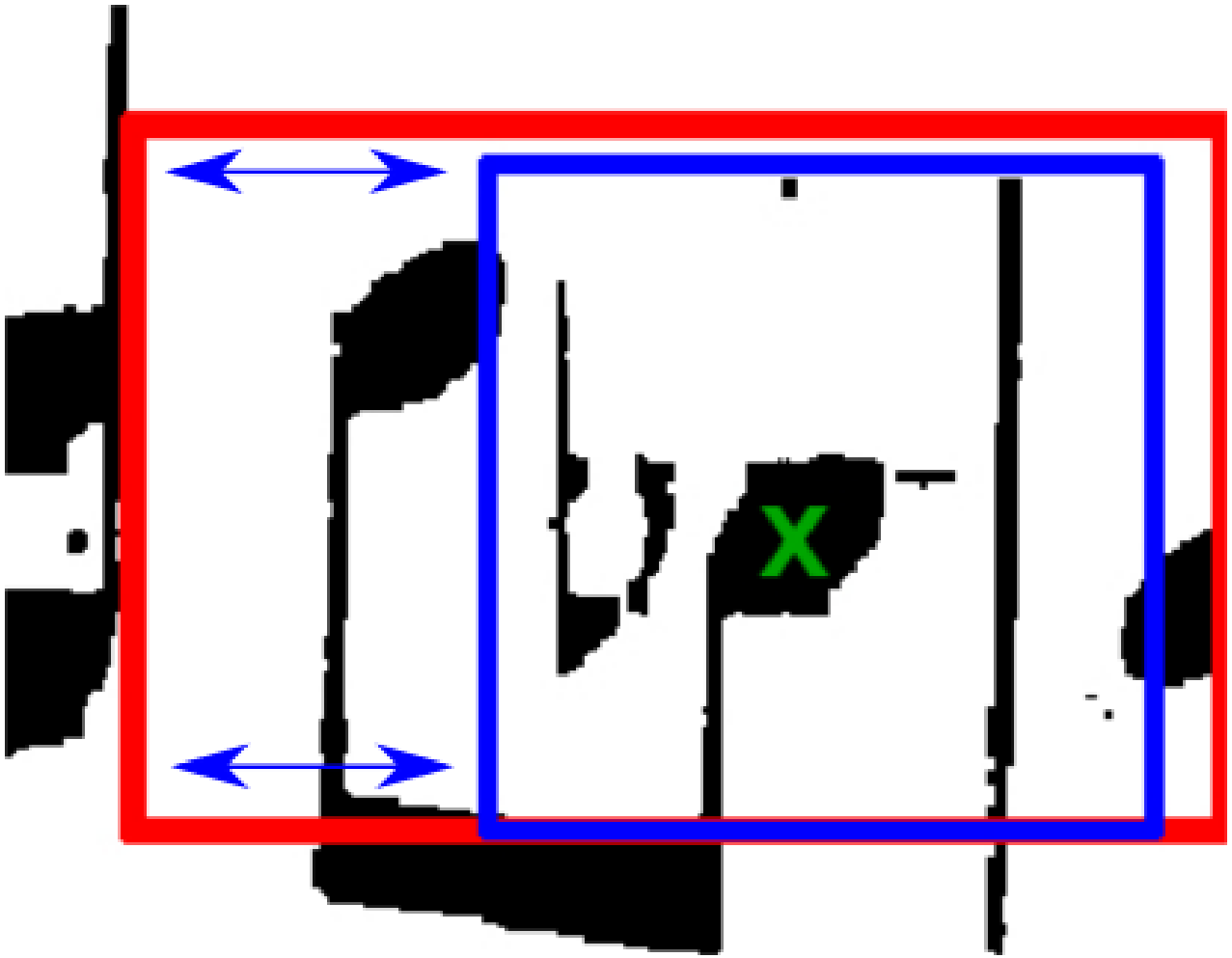


Figure: Random sampling on different zones

Bootstrapping

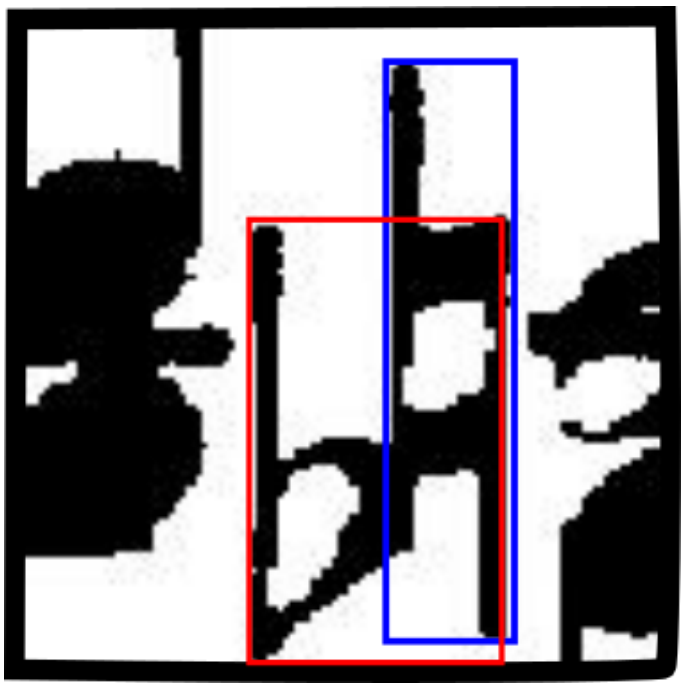
⇒ Generate **more data** for localization task.

From: 3k samples ⇒ to: 25k, 50k, 100k, 200k, 400k data!



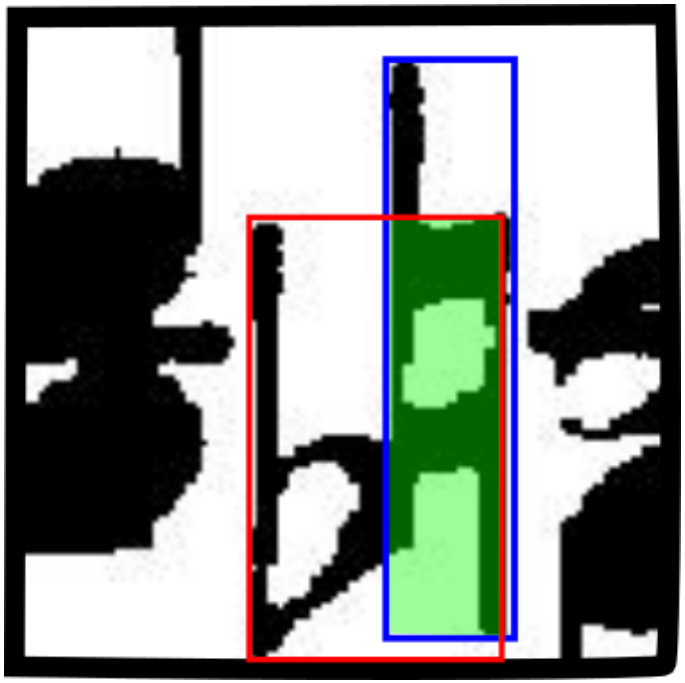
Intersection over Union

- How to evaluate this detection results?



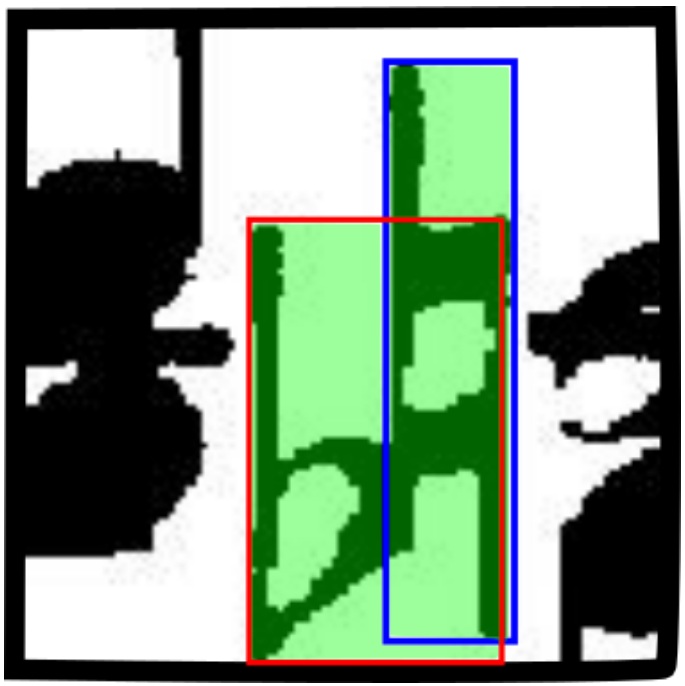
Intersection over Union

- How to evaluate this detection results?
  - Compute intersection area



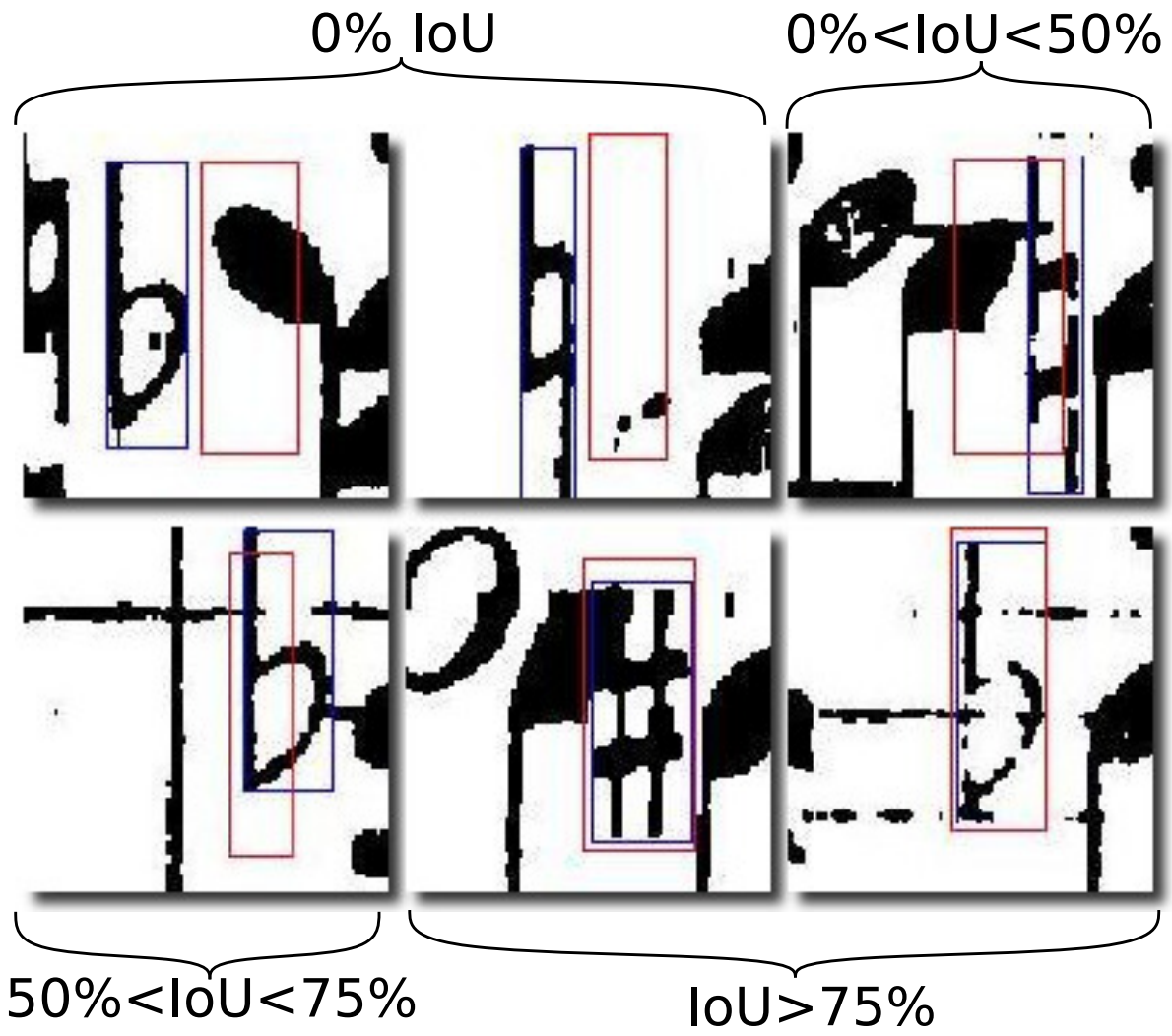
Intersection over Union

- How to evaluate this detection results?
  - Compute intersection area
  - Compute union area
- $IoU = \frac{Inter}{Union}$



Qualitative Results

- Blue is ground-truth
- Red is model prediction



Classic Detection Methods

- Spatial Transformer based detector [Choi, 2019]
- Region based detector like Faster R-CNN [Huang, 2016]
- ...

Results

Table: Results comparing the best Spatial Transformer (ST) based detector, Faster R-CNN, R-FCN and SSD [Huang, 2016]. Results shown are mAP (in %) with an IoU threshold of either 0.5 or 0.75.

Detectors	mAP with IoU >0.5		mAP with IoU >0.75	
	$\mu(\%)$	$\sigma(\%)$	$\mu(\%)$	$\sigma(\%)$
ST	97.25	1.68	94.81	2.99
Faster R-CNN	98.73	0.94	98.34	0.73
R-FCN	**99.17**	**0.30**	**98.73**	**0.40**
SSD	98.93	0.67	97.81	0.92

Plan

1 Background

2 Accidental Detection

3 SegGAN

4 Conclusion

Problem

Annotation Cost

- Manual annotation for object detection $\Rightarrow$ costly and time-consuming to produce
- e.g: MUSCIMA++ took 400 hours to produce

Proposal

- Reduce the need for manual data annotation
- Do detection task without explicit detection annotation
- At least generate enough detection annotation in order train a Faster R-CNN

Methodology

Classic Method

- Use synthetic data to train supervised detector
  - Generate synthetic music scores using music typesetting software
  - Train detector (like Faster R-CNN or SSD) using synthetic data
  - Apply trained detector on real not annotated dataset
- Pros:
  - Ability to generate massive amount of data using typesetting software
- Cons:
  - No guaranty that detector trained on synthetic data will work well on real data
  - Synthetic data generation is bound to music typesetting software (maybe there are no equivalent in other domains)

Methodology

Classic Method

- Use synthetic data to train supervised detector
  - Train detector (like Faster R-CNN or SSD) using synthetic data
  - Apply trained detector on real not annotated dataset
- Cons:
  - No guaranty that detector trained on synthetic data will work well on real data

Proposed Method

- Reduce the input size of images fed to detectors using DMOS grammar (reduce the search space)
- Bridge differences between synthetic data domain and real data domain using **Generative Adversarial Network (GAN)**

Generative Adversarial Network: A Min-Max Game

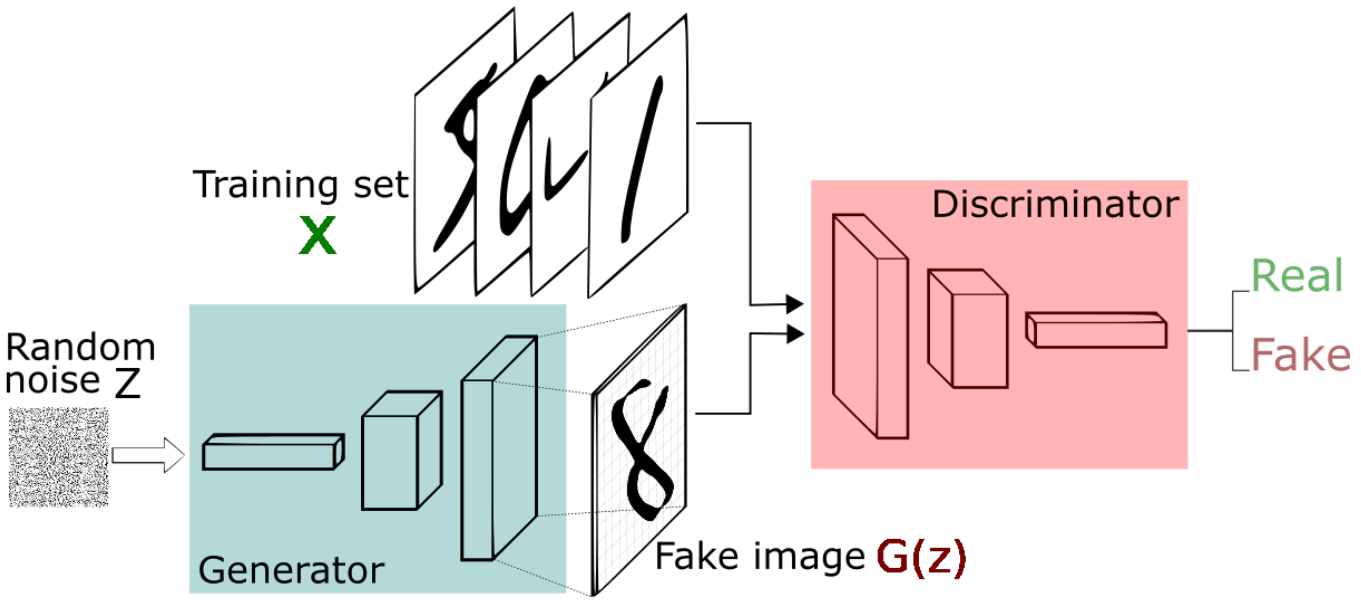


Figure: Generative Adversarial Network: Discriminator Mode

Discriminator Cost Functions to **Maximize**

$$\frac{1}{m}\sum_{i=1}^{m}[logD(x^i) + log(1 - D(G(z^{(i)})))]$$

Generative Adversarial Network: A Min-Max Game

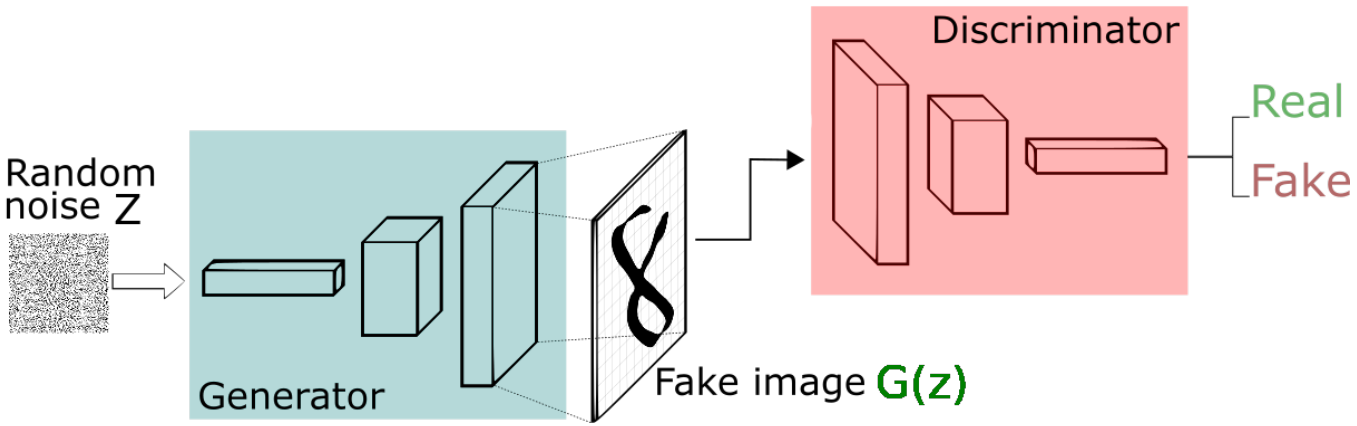


Figure: Generative Adversarial Network: Adversarial Mode

Adversarial Cost Functions to **Minimize**

$\frac{1}{m}\sum_{i=1}^{m} log(1 - D(G(z^{(i)})))$

Methodology

Key Concept

- Use **real isolated music symbol** from existing dataset to construct synthetic x training distribution
- Train a **Generator** to imitate synthetic data distribution
- Detect symbols in images generated by **Generator** using **SSD detector** pre-trained on synthetic data

Segmentation GAN

- Replace Generator with a U-Net network [Ronneberger 2015] [Luc 2016]

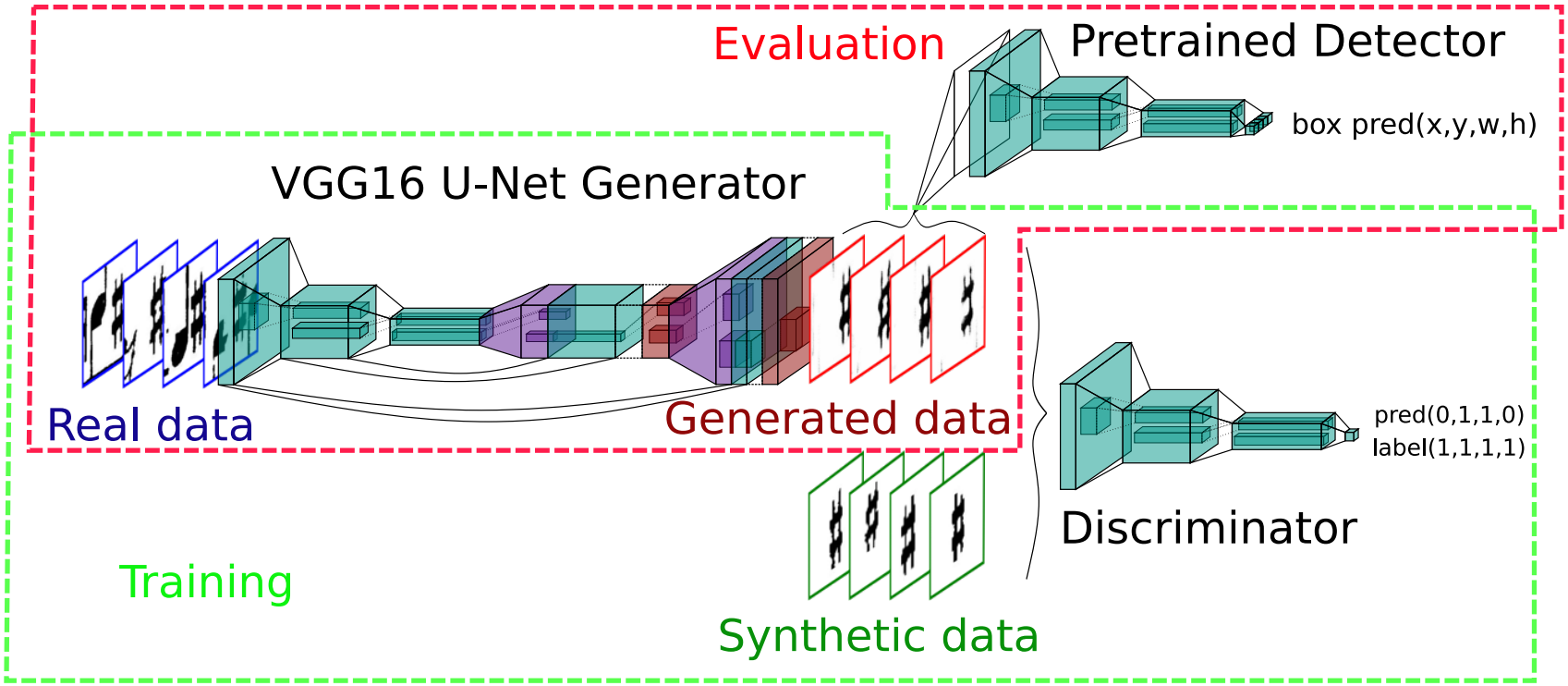


Figure: SegGan + pre-trained detector

Segmentation GAN

- Replace Generator with a U-Net network [Ronneberger 2015] [Luc 2016]

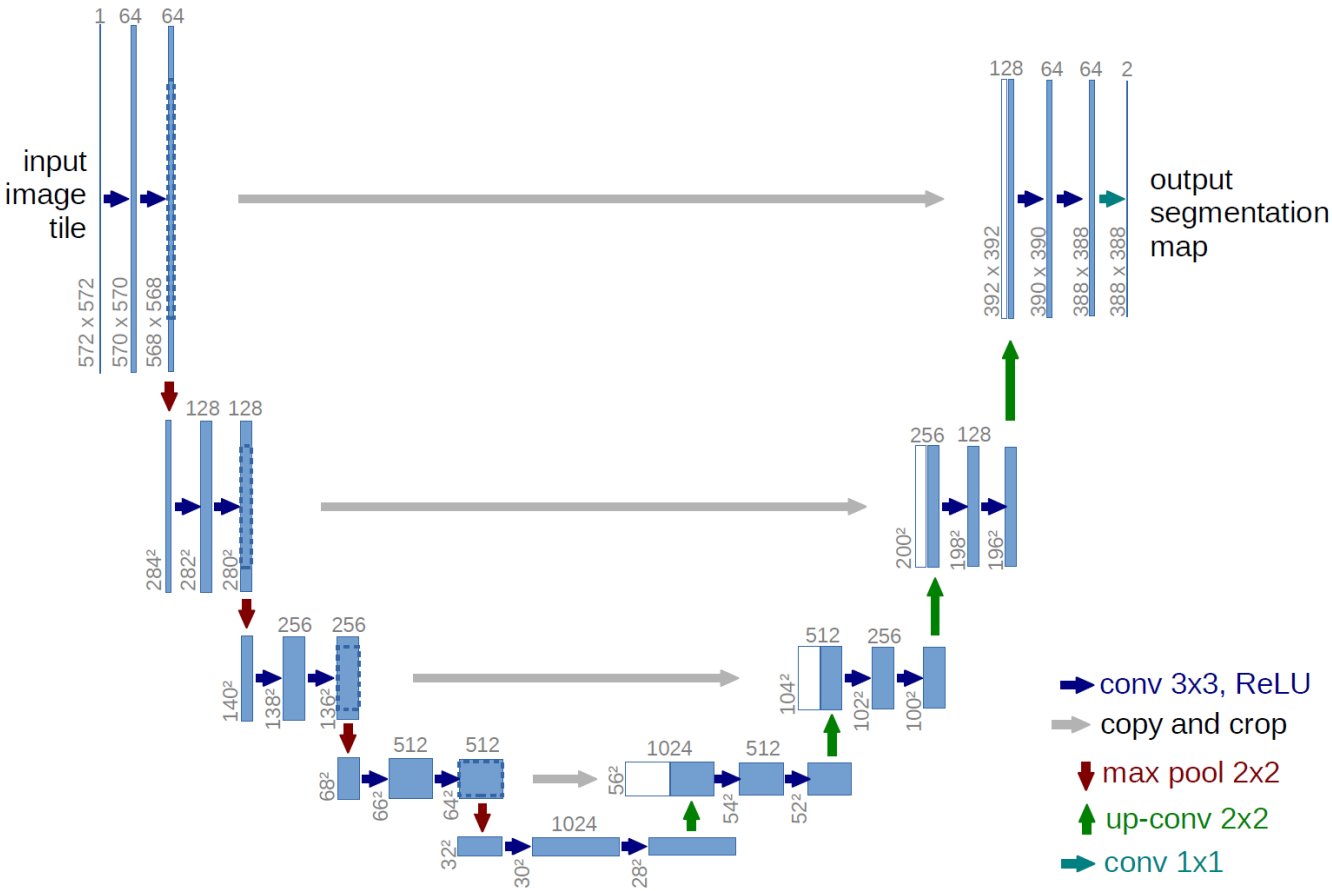


Figure: Using a U-Net for the generator

Segmentation GAN

- Replace Generator with a U-Net network [Ronneberger 2015] [Luc 2016]

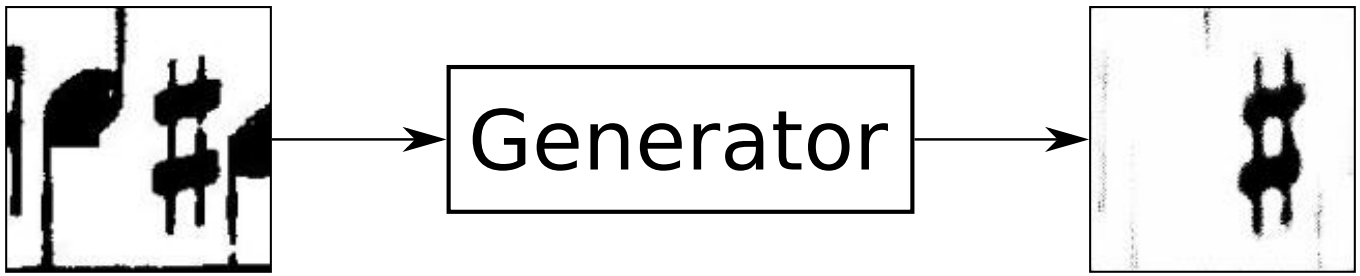


Figure: Generator should output a mask extracting symbol

Segmentation GAN

- Replace Generator with a U-Net network [Ronneberger 2015] [Luc 2016]

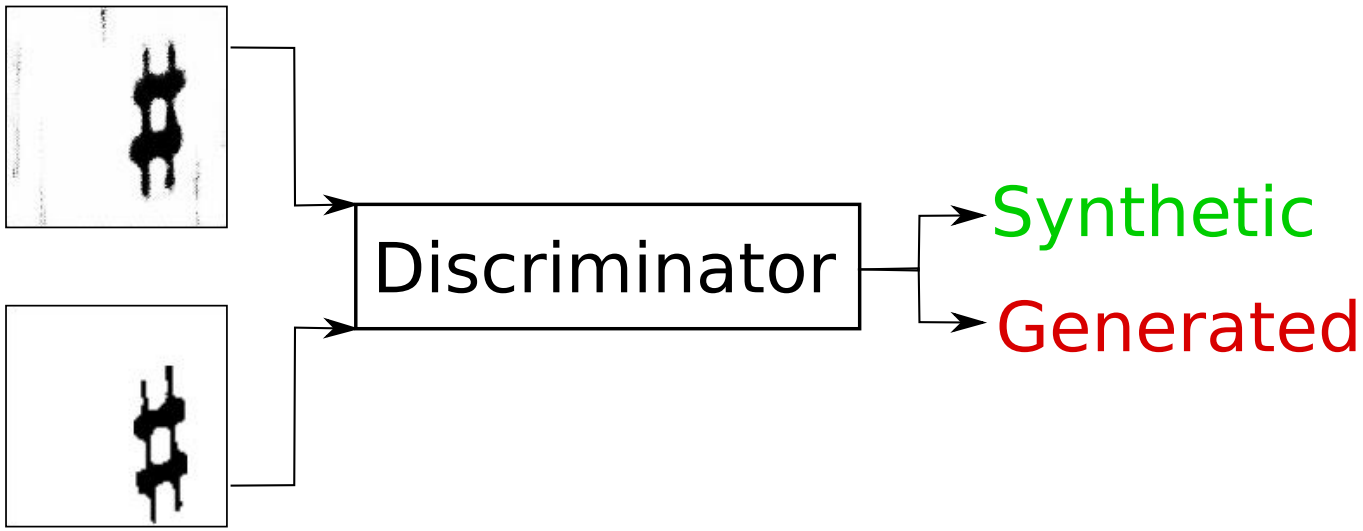


Figure: Discriminator should discriminate between mask and synthetic image

Using Isolated Music Symbol Dataset

- Replace ideal x distribution with synthetic data
- Random position and scale using Uniform Distribution
- Arbitrarily set min/max width/height sizes

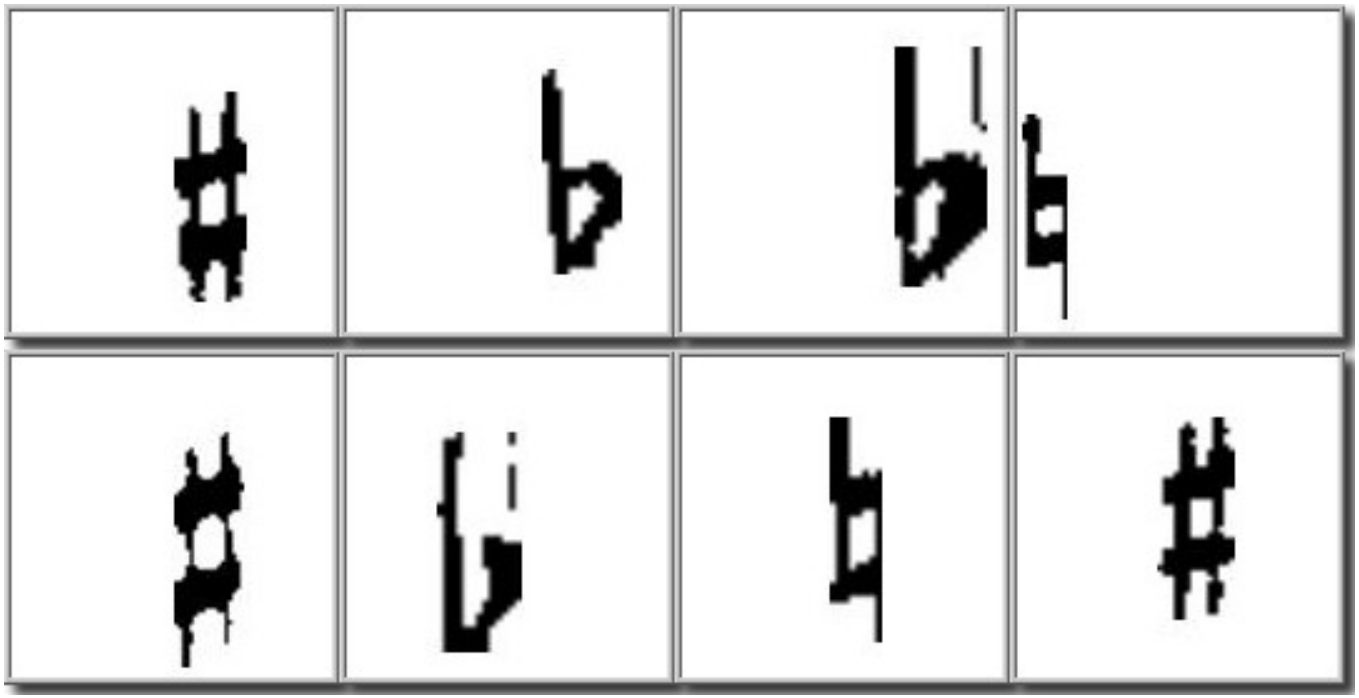


Figure: Example of synthetic images generated from *real* isolated symbols.

Supervised Generator Loss

Proposition

- Improve training of **Generator** using additional loss with synthetic data
- Learn identity transform for symbol detection
- Learn rejection task
  - Use non-target class of isolated symbols as rejection examples
  - Define rejection as the task of outputting a blank image

Supervised Generator Loss

- **Important**: keep the balance in training between Discriminator and Generator

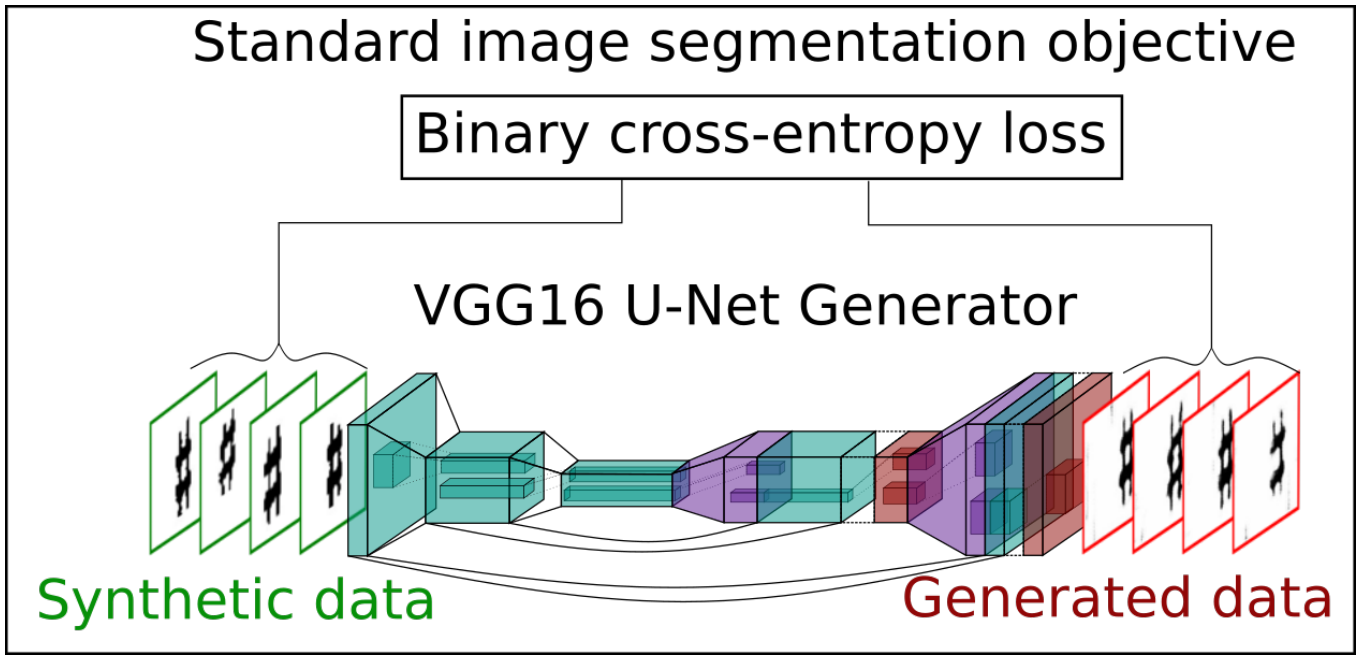


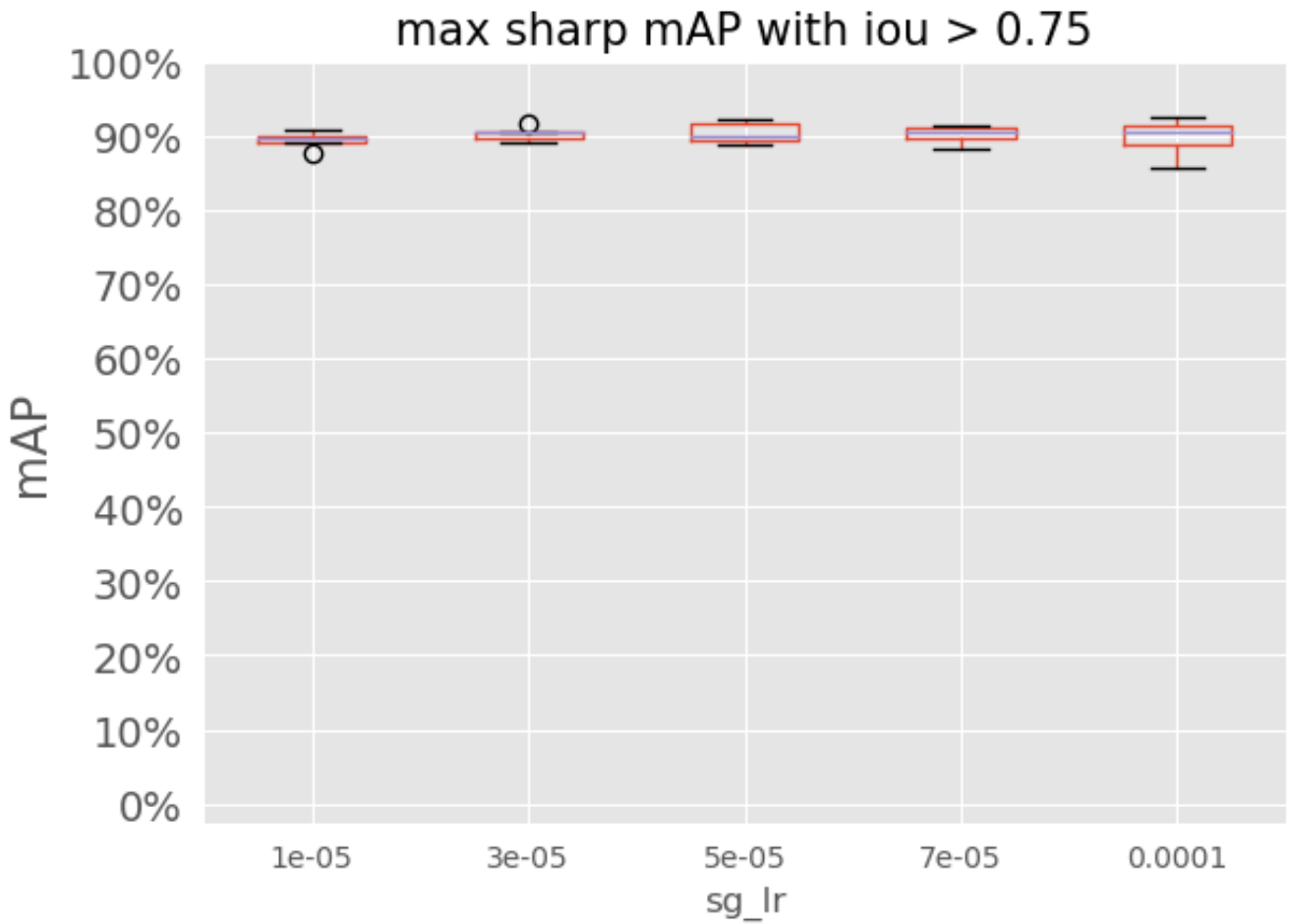
Figure: Supervised Generator Loss

Results

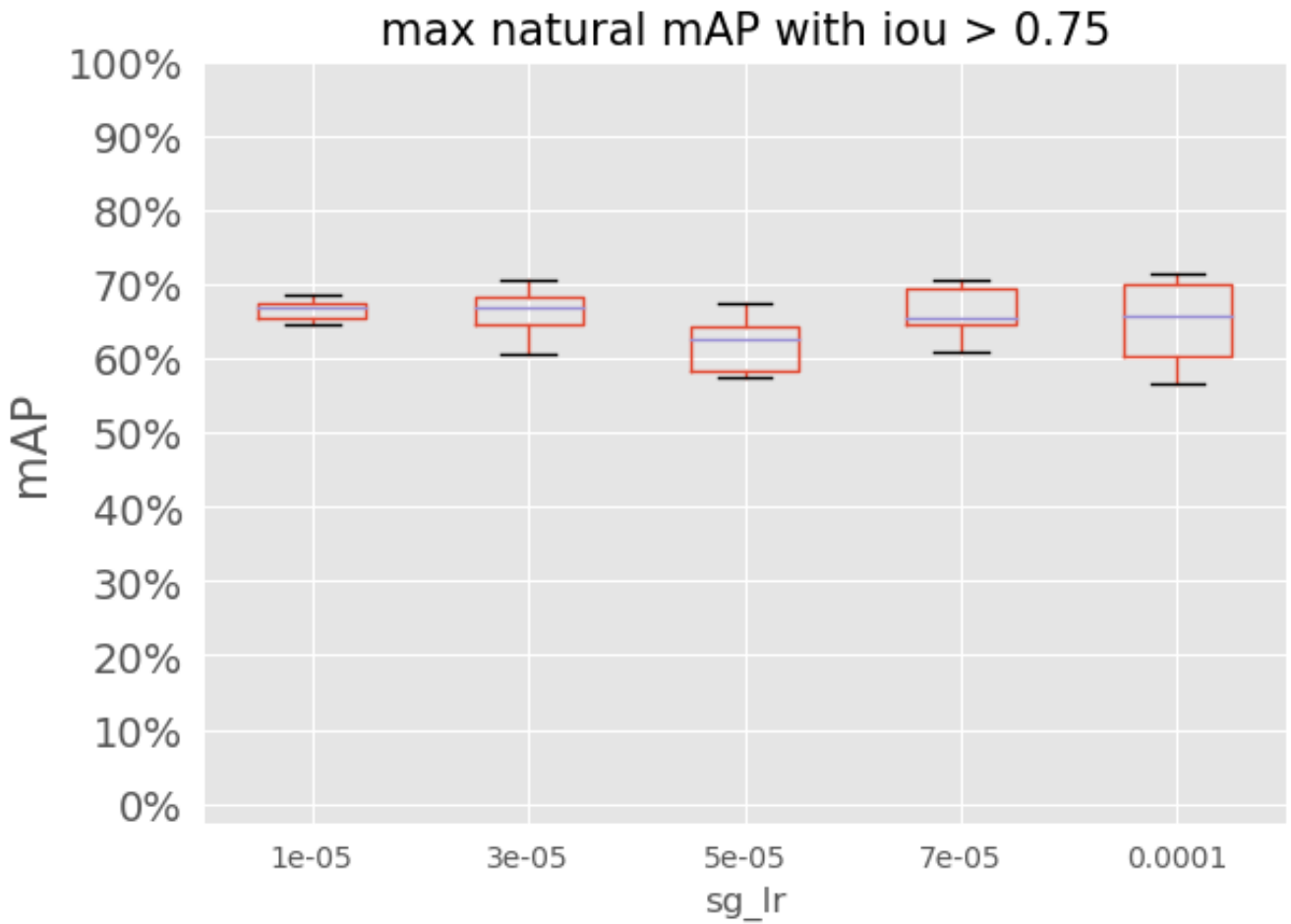
Experimental conditions

- RMSProp optimizer
- Learning rates balancing Generator and Discriminator training:
  - discriminator lr: 0.0002
  - adversarial lr: 0.0001
  - supervised generator loss lr: gridsearch between [1e-5, 1e-3]
- Apply pre-trained SSD detector on generator image
  - Evaluate results using mAP with IoU $>$0.75 metric

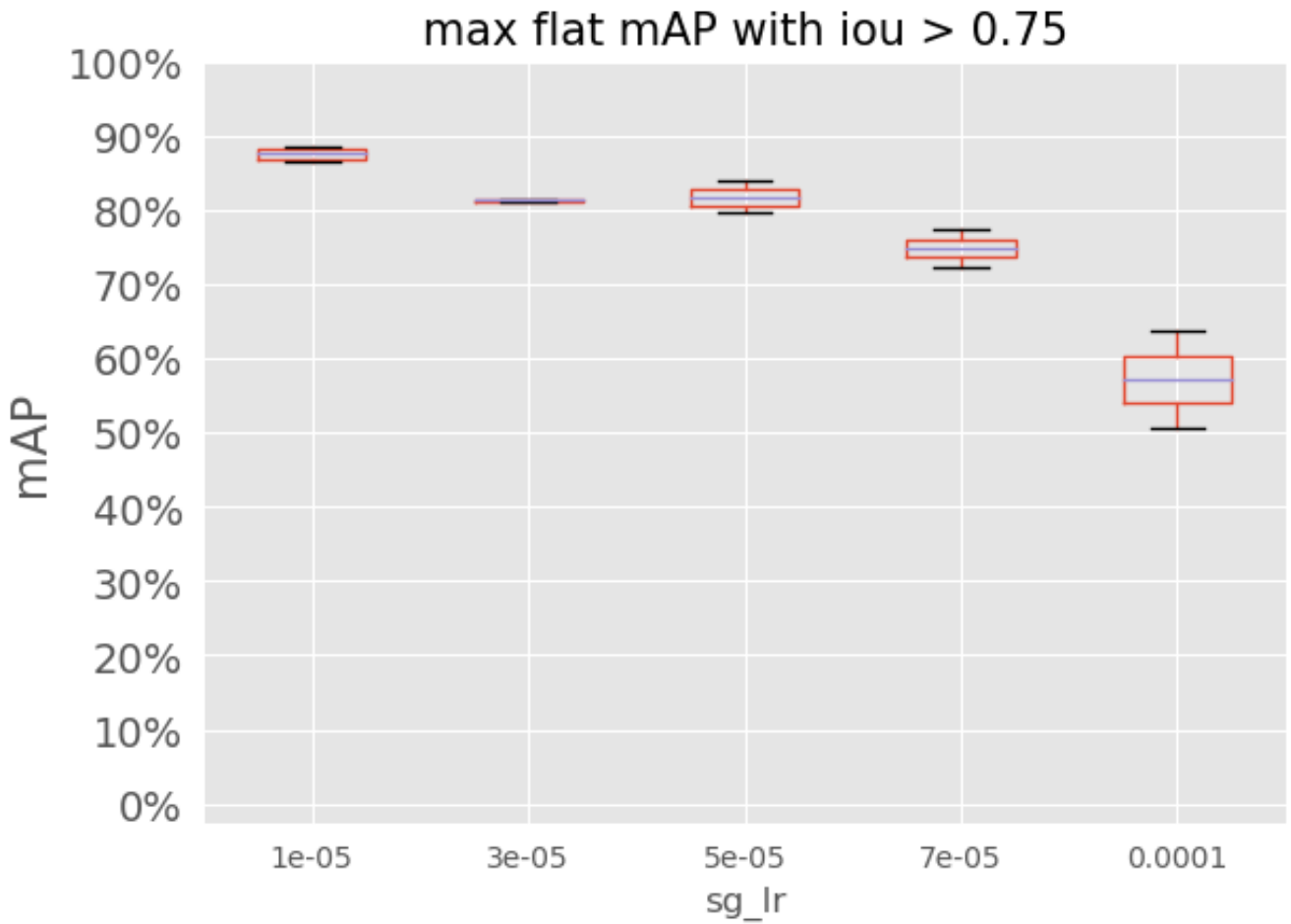
Results


max sharp mAP with iou > 0.75
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
mAP
1e-05
3e-05
5e-05
7e-05
0.0001
sg_lr

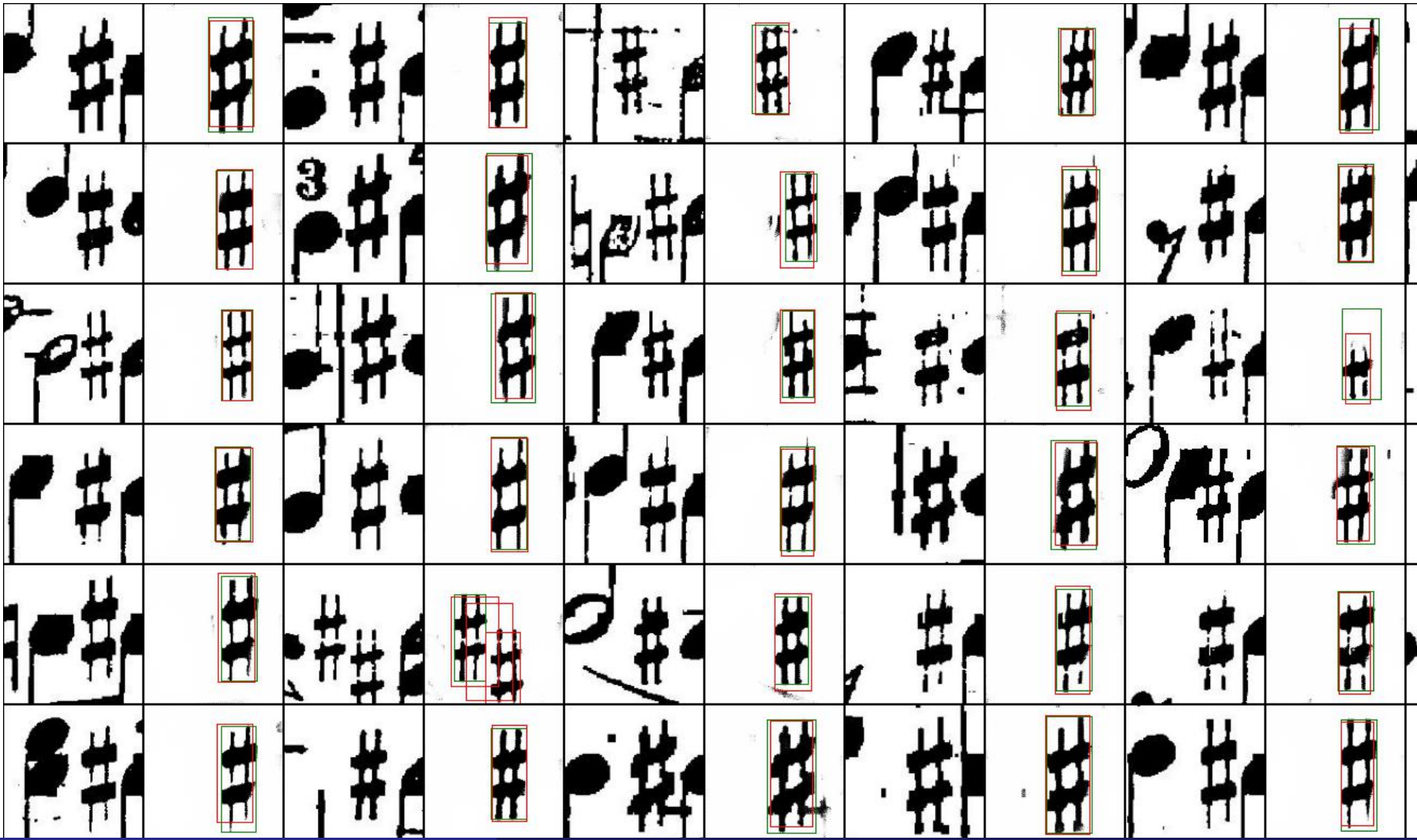
Results


max natural mAP with iou > 0.75
mAP
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
1e-05
3e-05
5e-05
7e-05
0.0001
sg_lr

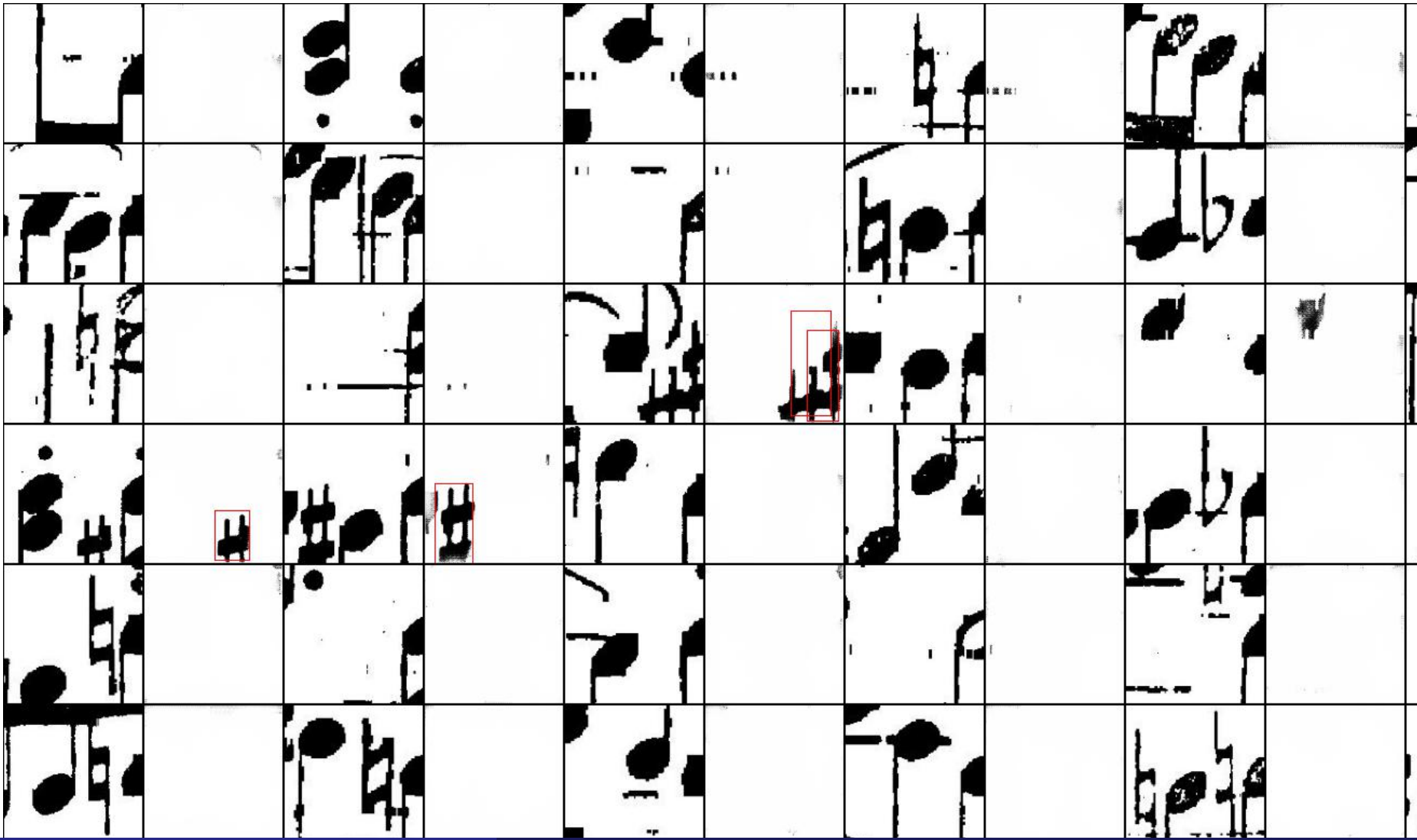
Results


max flat mAP with iou > 0.75
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
mAP
1e-05
3e-05
5e-05
7e-05
0.0001
sg_lr

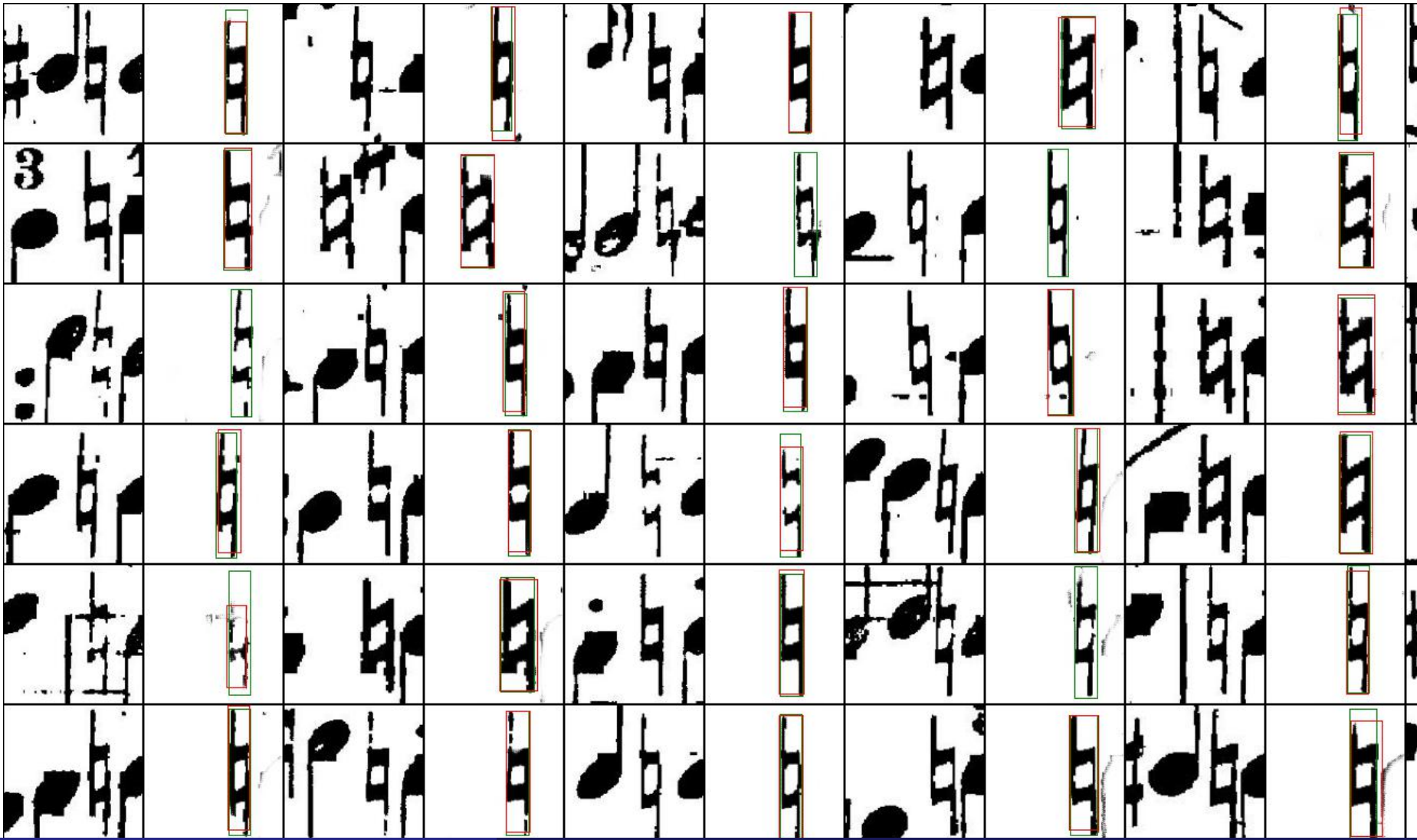
Qualitative Results



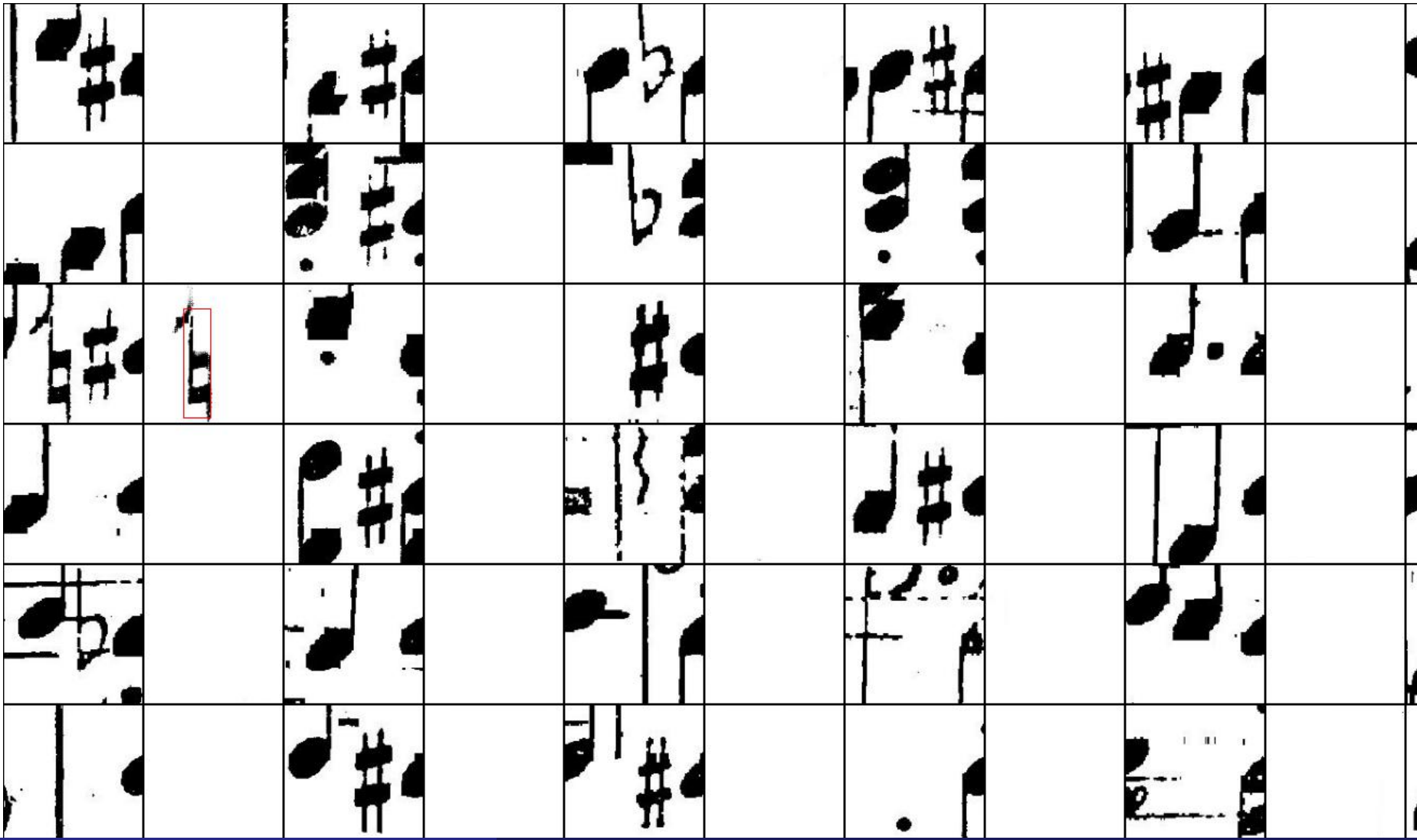
Qualitative Results



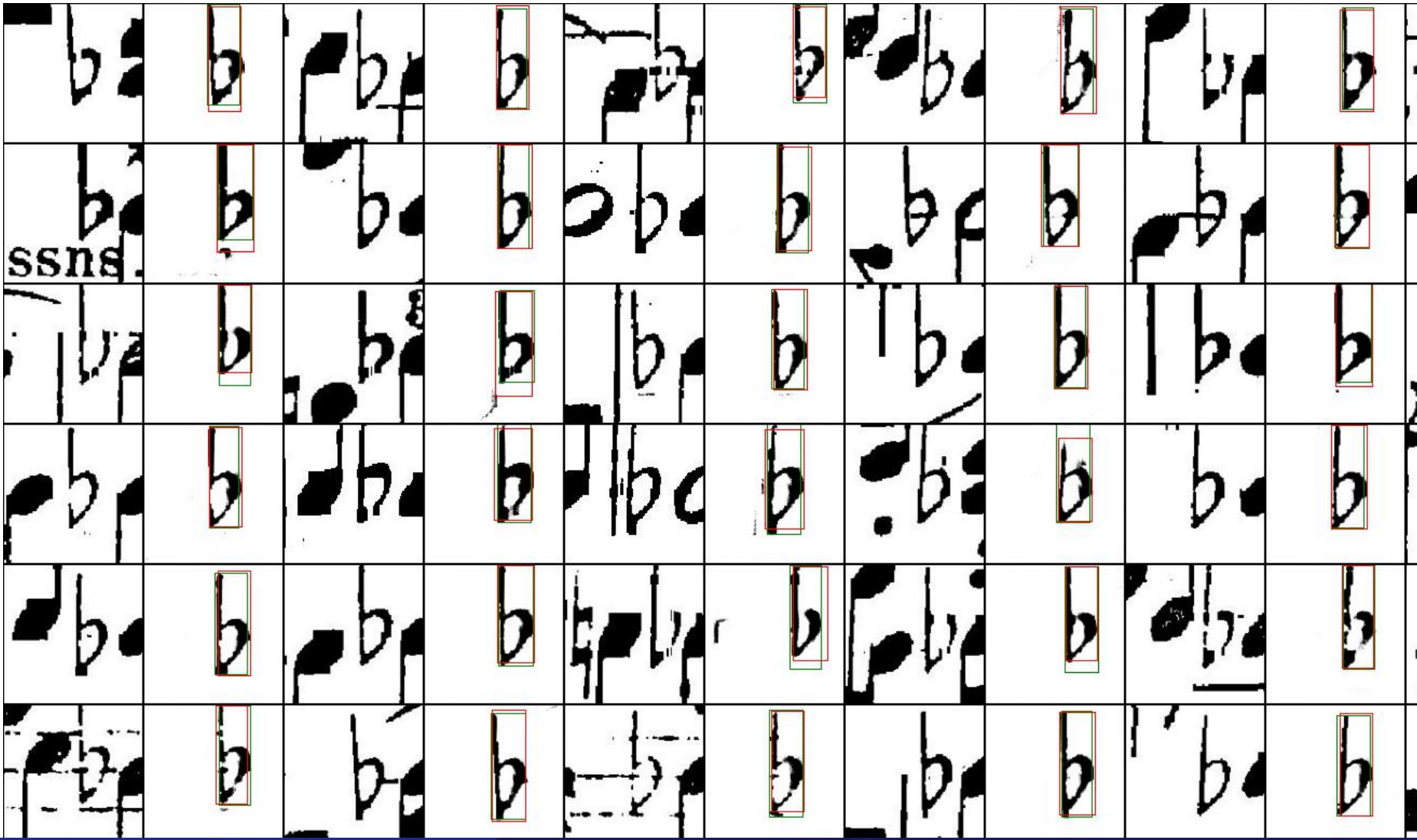
Qualitative Results



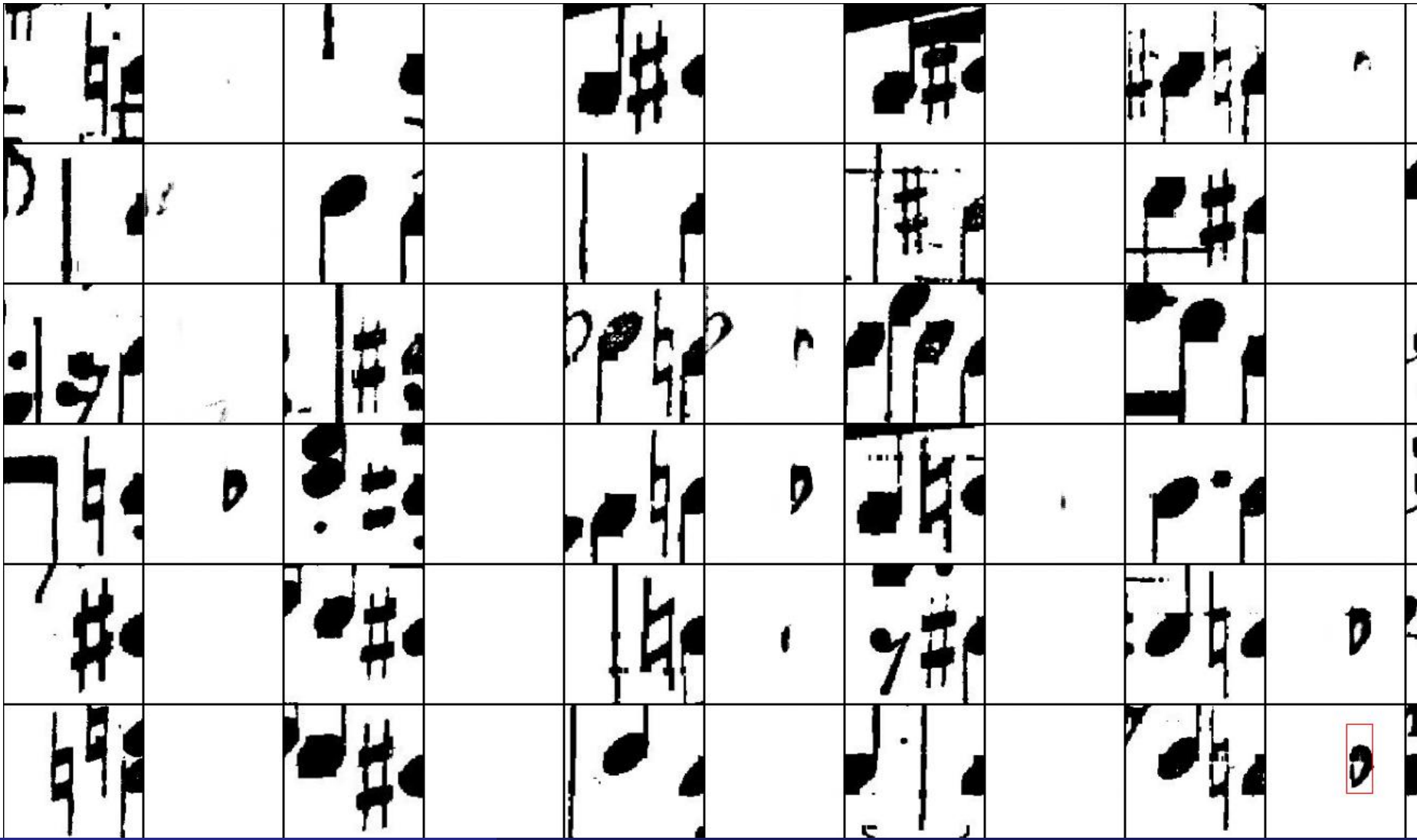
Qualitative Results



Qualitative Results



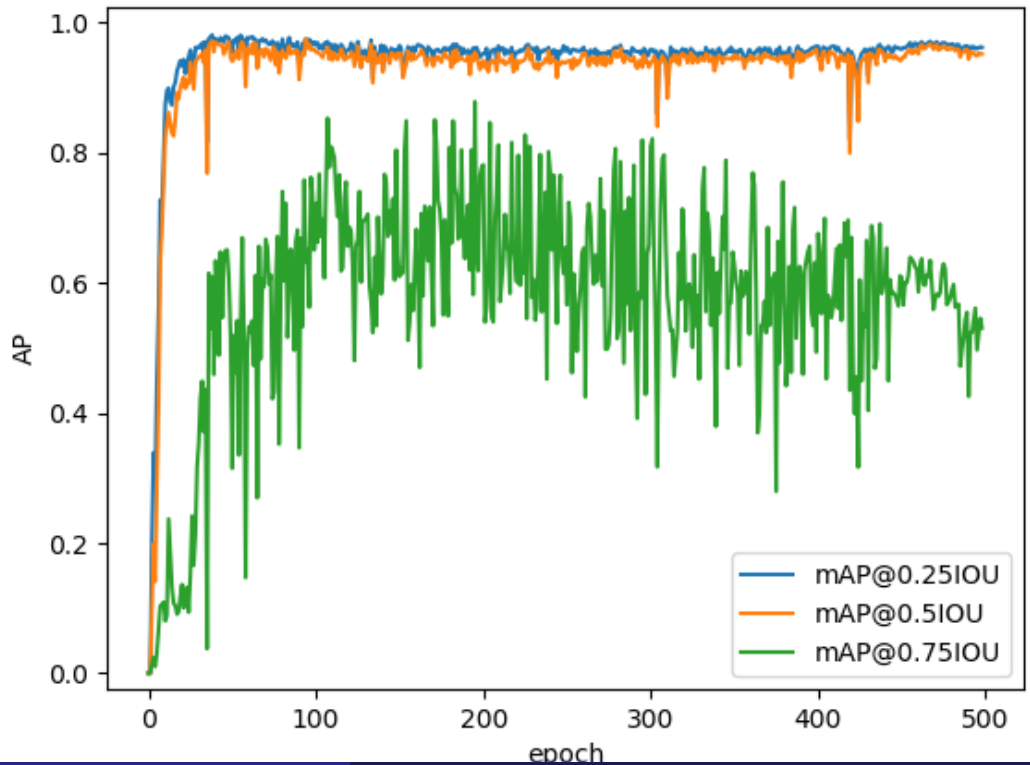
Qualitative Results



Early Stopping Problem

Instability of GAN training

- GAN training is an oscillation around an equilibrium point



Early Stopping Methodology

Goal

- Find optimal GAN state during training that gives best mAP results

Possible Solutions

- Use synthetic data
  - Current generated synthetic not suitable for early stopping (data is too simplistic)
  - Use synthetic data generated from music typesetting software
  - Possibility to vary iou threshold to better discriminate GAN state
- Use a very small validation set (20/50 examples per class) of manually annotated symbols in real data

Summary

Goal

Train Deep-Learning detectors for music symbol detection using only shape information of an isolated music symbol dataset.

Methodology

- Generate synthetic data using isolated music symbols
- Use SSD detector trained on synthetic data
- Bridge gap between real data and synthetic data using GAN
  - Guide/improve GAN training by designing synthetic data generation strategy and additional loss

Plan

1 Background

2 Accidental Detection

3 SegGAN

4 Conclusion

Conclusion

Overview

- Use GAN model for unsupervised detection of music symbol
  - Bridge differences between synthetic data and real data using **Generative** model
  - Do detection task without explicit detection annotation (use only shape information from isolated symbols)
  - Reduce the need for manual data annotation

Future Work

- Early stopping of the GAN training
- Improve SegGan model for multi-symbol detection
- Extend application of SegGan to more symbol classes

References

Ronneberger et al. (2015)

U-Net: Convolutional Networks for Biomedical Image Segmentation.

In Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, 234–41. Lecture Notes in Computer Science

Luc et al. (2016)

Semantic Segmentation Using Adversarial Networks.

Huang et al. (2016)

Speed/Accuracy Trade-Offs for Modern Convolutional Object Detectors

Choi et al. (2019)

CNN-Based Accidental Detection in Dense Printed Piano Scores.

In 2019 15th IAPR International Conference on Document Analysis and Recognition (ICDAR). Sydney, Australia