Handwritten-notes Denoising

Authors

Olmo Baldoni, Cristian Bellucci, Danilo Caputo

325524@studenti.unimore.it , 322906@studenti.unimore.it, 246019@studenti.unimore.it

Introduction

- **Challenge:**
  - Handwritten notes, especially on lined or squared sheets, are intrisacally noisy for automatic extraction of text and equations.
- **Solution:**
  - Train a Convolutional Neural Network (CNN) to remove square grids from handwritten notes, resulting in a text-only, noise-free image.
- **Methodology:**
  - Utilize UNet
  - Create synthetic dataset for the training process
- **Additional Tools:**
  - An approach to crop and warp images of hand-taken notes from photos.
  - A retrieval system for automating the collection of images to be processed by the denoising network.

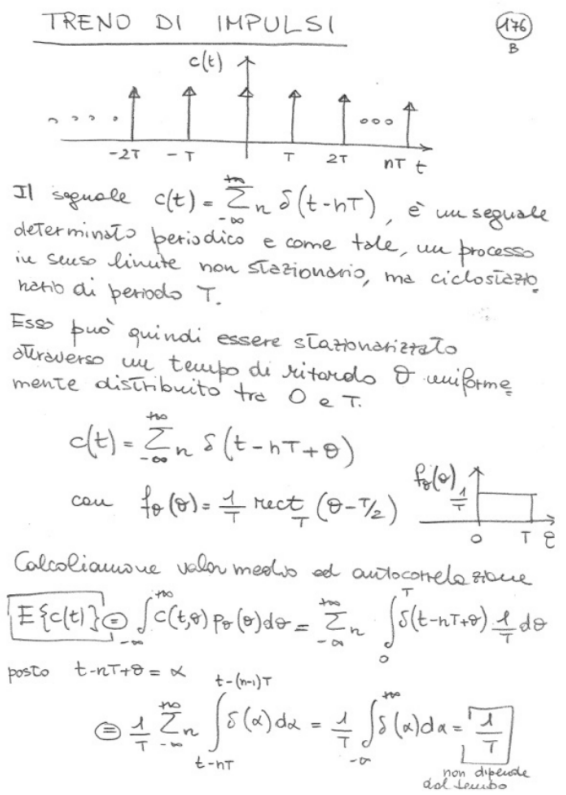
Synthetic Dataset

- 132000 synthetic images obtained by a Data Augmentation process.
- 3300 images of handwritten notes on blank sheets
- 70 different template of different grids used as background
  - 40 of each randomly applied

- The grids G(x, y) were overlapped to the notes images N(x, y) using the max operator over the negative gray scale images:

$$I_{neg}(x, y) = \max(G_{neg}(x, y), N_{neg}(x, y))$$

$$I(x, y) = 255 - I_{neg}(x, y)$$



(a) Example of handwritten note

(b) Example of template

Data Augmentation

- Making synthetic images more realistic

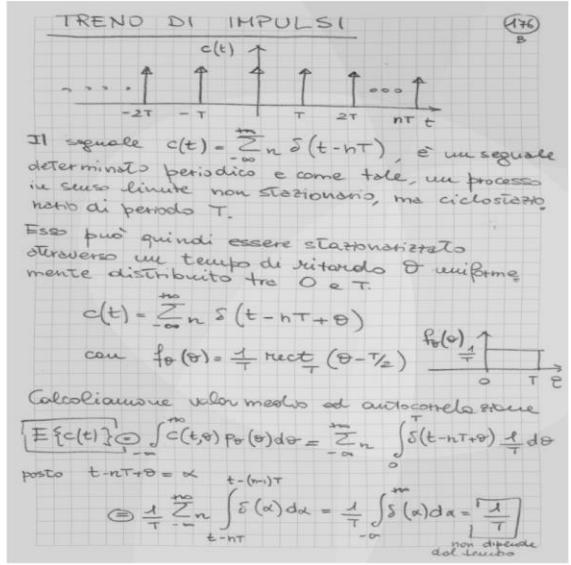
Applied:

- Sinusoidal distortions that make the pixels of the templates slightly wavy

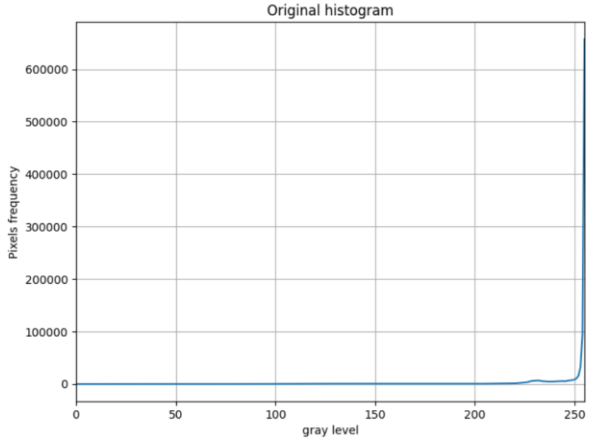
$$x_{new} = x + A \sin \frac{y}{k}$$
$$y_{new} = y + A \sin \frac{x}{k}$$

- Elliptical masks randomly applied to one or more regions of the image to alter the brightness

(a) Image without brightness changes



(b) Image with brightness changes



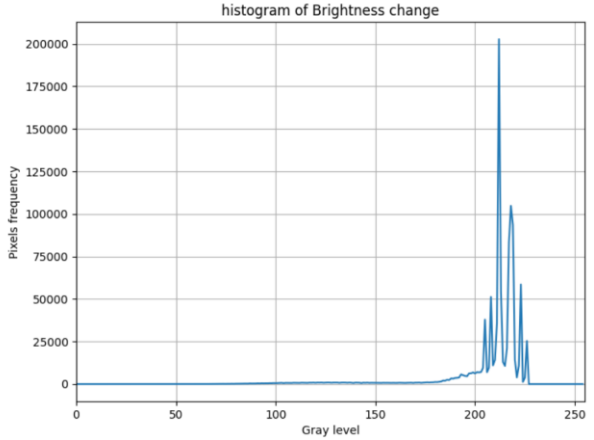


Image cropping and warping

- Crop a sheet of paper in a image and warping it to original resolution.

First step:

- detecting sheet contours
  - morphological operation combined with Canny algorithm.

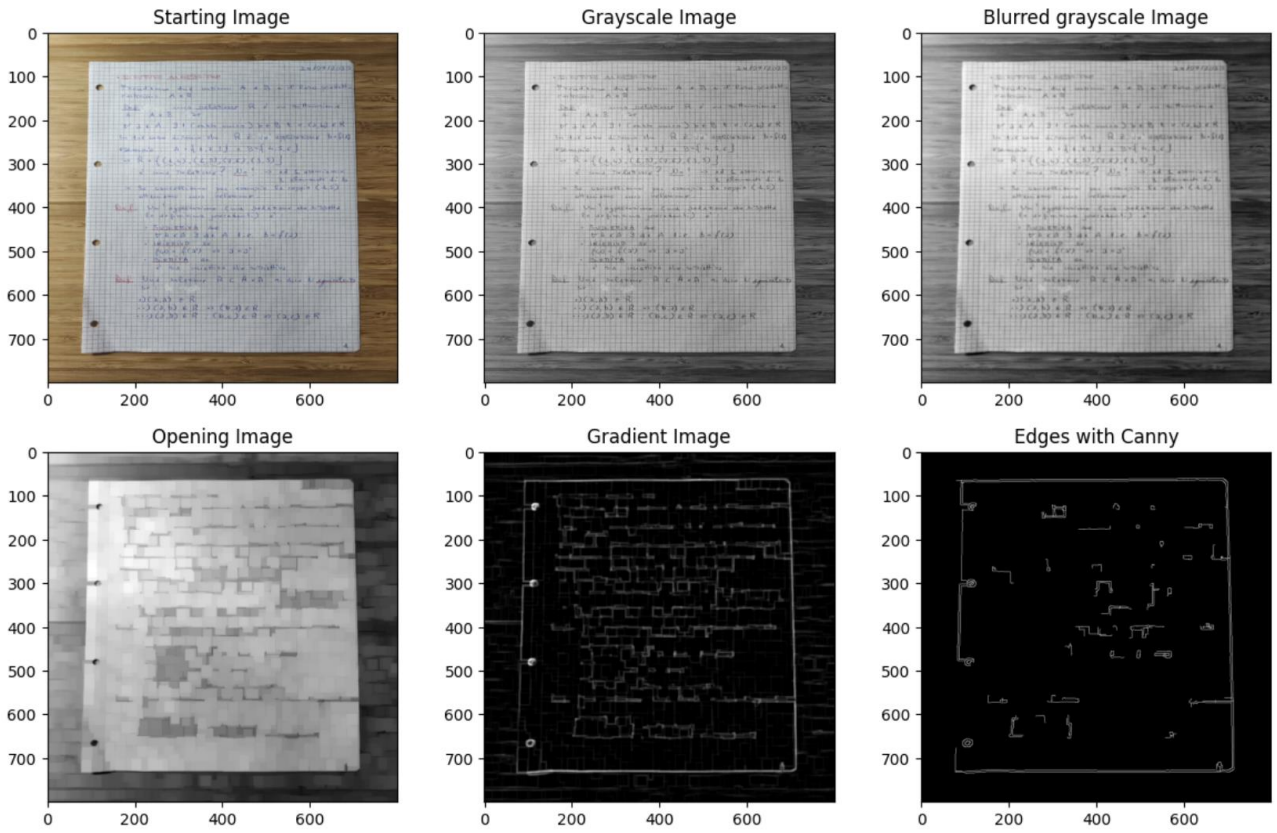


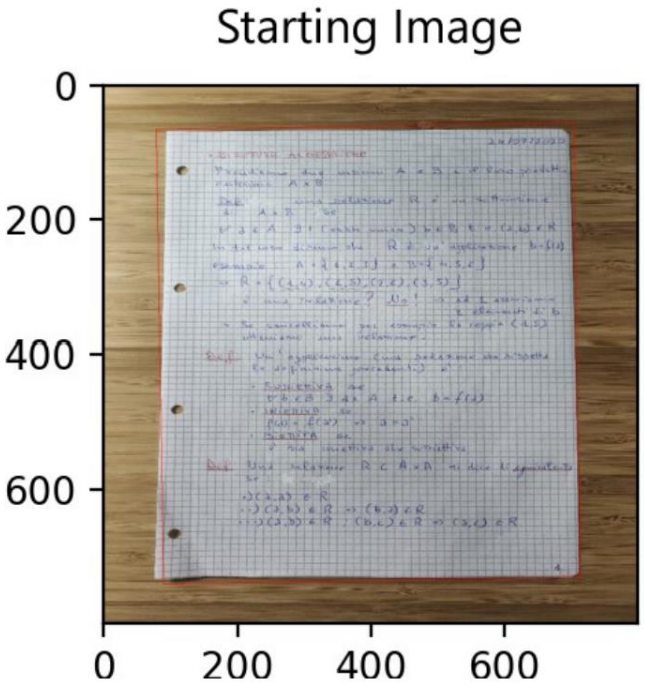
Figure 4: Pipeline of edge detection

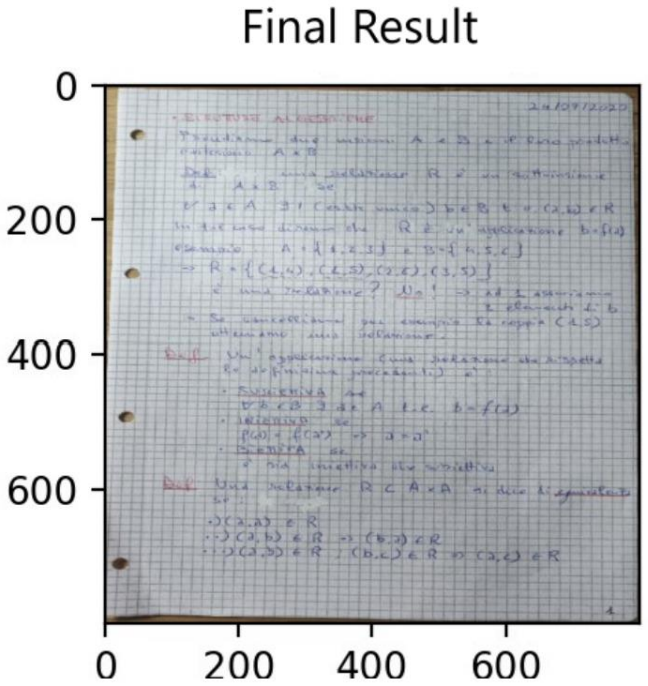
Image cropping and warping

- Crop a sheet of paper in a image and warping it to original resolution.

Second step:

- Perspective Transformation
  - Extraction of corners coordinates of contour with maximum area
  - Map the detected sheet to corner of original image

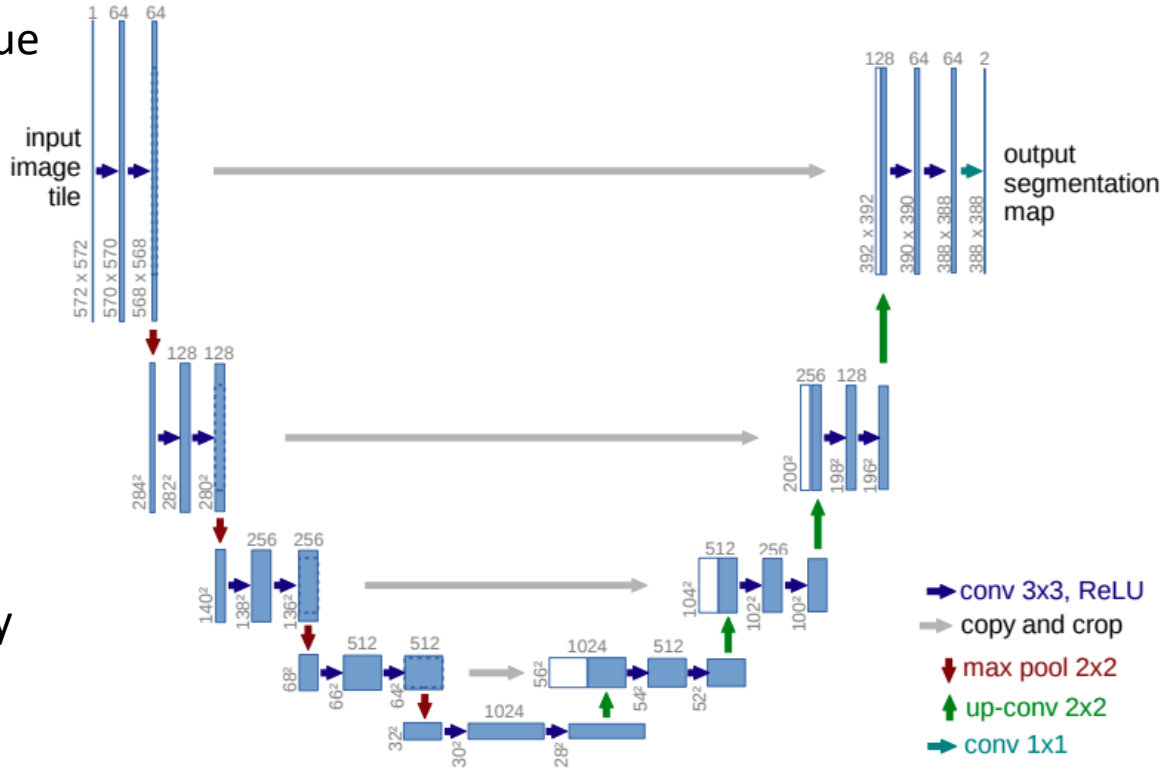




UNet

Choosing UNet for Grid Removal

- Previous methods like Fourier Transform were impractical due to:
  - Manual parameter tuning
  - Lack of automation
  - time consumption
- The need for a scalable and generalized solution.
- Switched to a deep learning approach using UNet for grid removal.
- UNet chosen due to its effectiveness in segmentation, ability to downsample and upsample.
- Suitable for complex images, and has the ability to maintain clarity of text and fine details.



UNet

Train&Test set Composition

- Image resolutions tested: 256, 512, and 1024 pixels
- Dataset divided into 2/3 train and 1/3 test split
- Overfitting issue:
  - Model primarily learning to remove synthetic grids
- Addressing overfitting:
  - Increased number of grids
  - data augmentation
  - expanded base images and grid variety

UNet

Training Approach and Optimizers

- Distributed training to handle large images efficiently.
- Use of Mean Squared Error (MSE) loss function.
- Optimal settings: Batch size of 2 per process, initial learning rate of 0.01, dynamic learning rate adjustment with maximum learning rate of 0.3.

- SGD and Adam optimizers: SGD used with onecycle scheduler and momentum of 0.9.
- Selection of the best model based on validation loss.

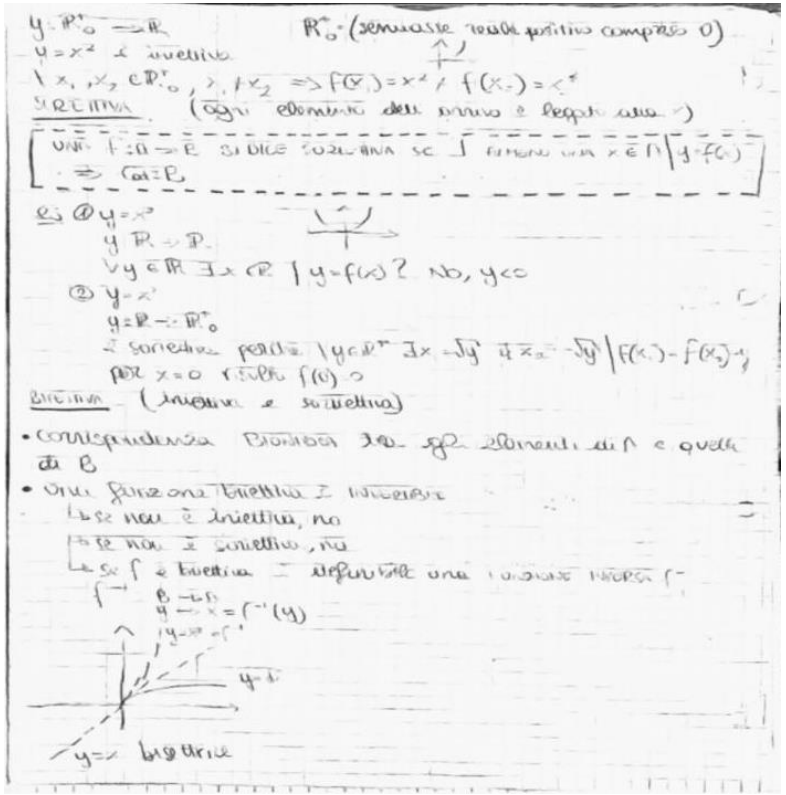


Figure of a real image inferenced

UNet

Experimental Methods

- Evaluation metrics used: Mean Squared Error (MSE) and Structural Similarity Index (SSIM).

- MSE: Measures the average of squared differences between the predicted (denoised) and the ground truth images.

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(\hat{I}(i) - I(i)\right)^2$$

- SSIM: Compares the similarity between two images, using an uncompressed or distortion-free image as a reference.

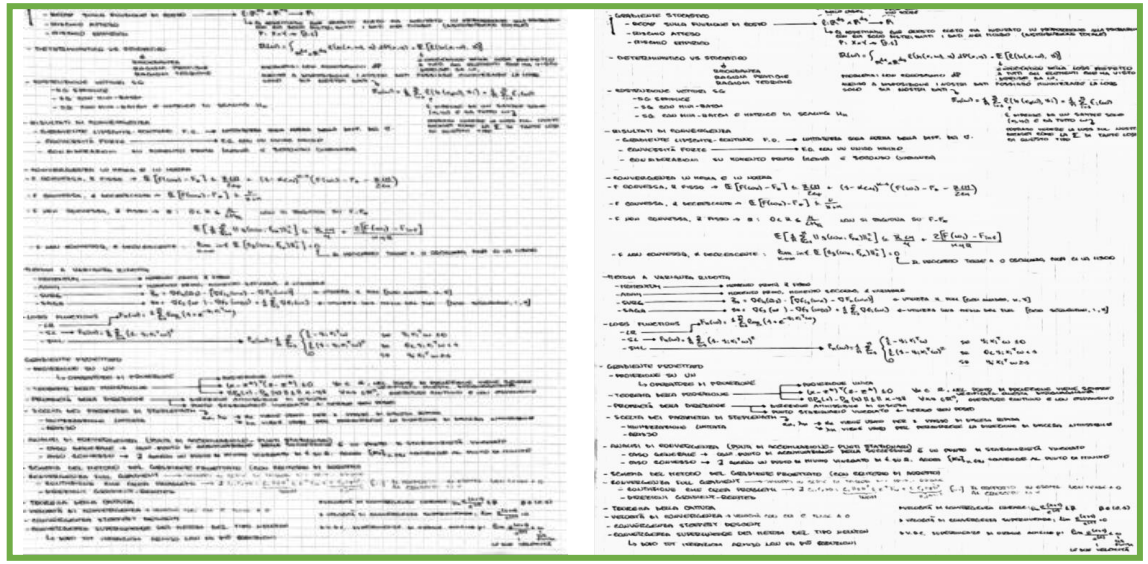
$$\mathrm{SSIM}(\hat{I}, I) = \frac{(2\mu_{\hat{I}}\mu_I + C_1)(2\sigma_{\hat{I}I} + C_2)}{(\mu_{\hat{I}}^2 + \mu_I^2 + C_1)(\sigma_{\hat{I}}^2 + \sigma_I^2 + C_2)}$$

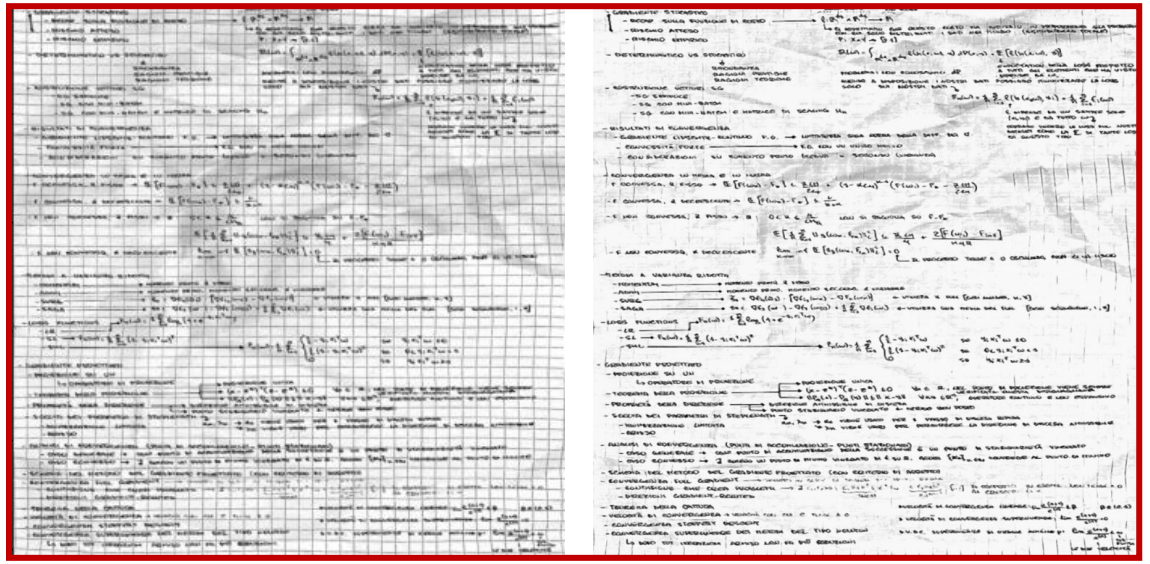
UNet

Evaluations and Results

- The table presents the difference between the denoised images and the ground truth images, contrasted against the synthetically applied grid images and GT.
- Observation: Although evaluation methods like MSE and SSIM provide quantitative results, visual inspection remains the most effective way to assess model efficiency.

Figure	MSE		SSIM	
	Inferenced	Original	Inferenced	Original
22	0.0580	0.0063	0.4672	0.6743
23	0.0691	0.0981	0.4424	0.1724
24	0.0582	0.0924	0.4745	0.1598

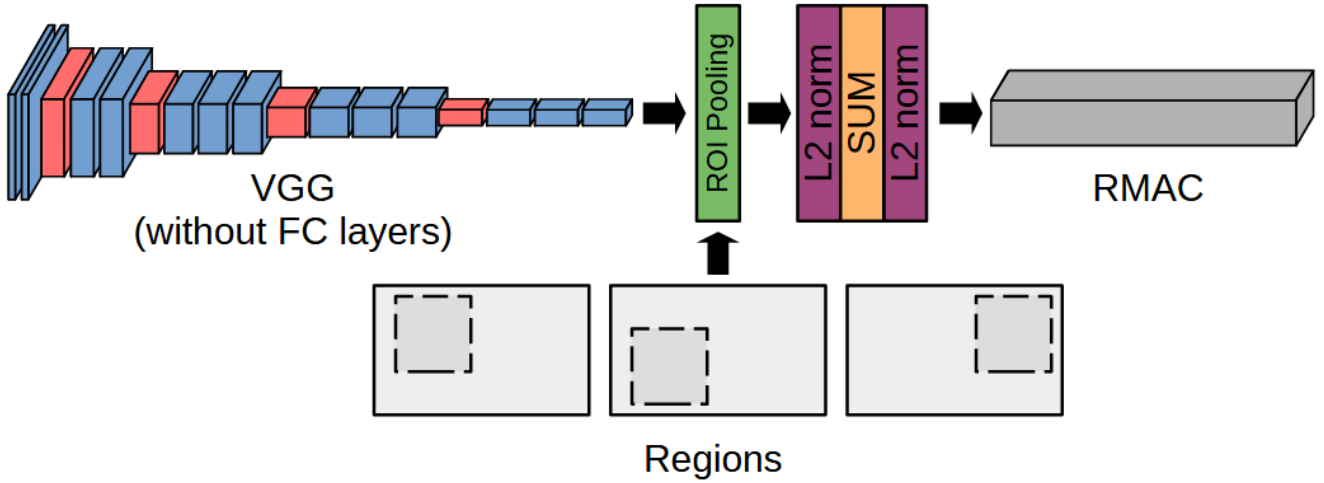




CBIR for Handwitten Notes

Image Retrieval Pipeline

1. Input image is preprocessed and converted to tensor
2. Tensor is passed through pretrained CNN feature extractor
3. Apply RMAC on activations to get regional vectors
4. Vectors normalized and summed into global descriptor
5. Descriptor compared to database to retrieve top k matches



CBIR for Handwitten Notes

Computing the RMAC Vector

- *MAC* vector takes max activation per channel

$$f = [f_1, ..., f_k, ..., f_K]^T, \quad \text{with} \quad f_k = \max_{x \in X_k} x$$

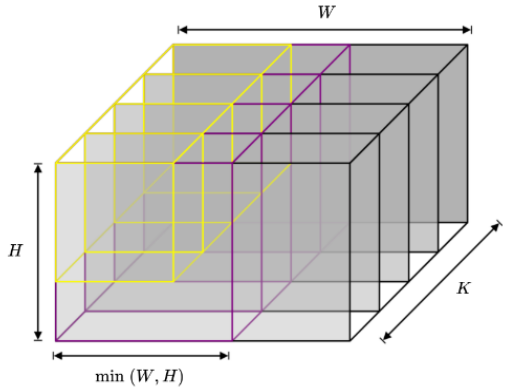
- *RMAC* divides tensor into overlapping regions

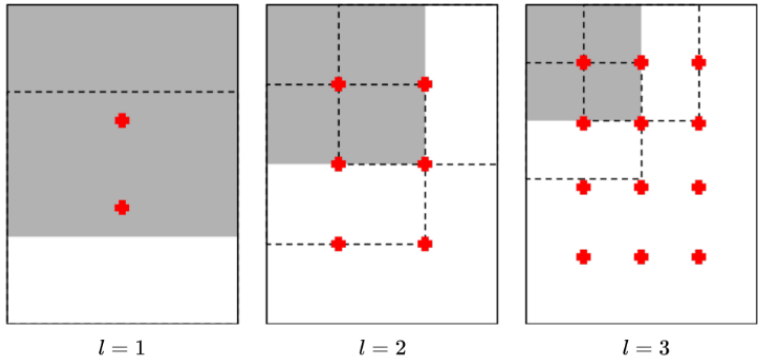
$$f_{R_i} = [f_{R_i,1}, ..., f_{R_i,k}, ..., f_{R_i,K}]^T, \quad \text{with} \quad f_{R_i,k} = \max_{x \in R_{i,k}} x$$

- Regional MAC vector computed per region

$$F = \sum_{i=1}^{N} f_{R_i} = [\sum_{i=1}^{N} f_{R_i,1}, ..., \sum_{i=1}^{N} f_{R_i,k}, ..., \sum_{i=1}^{N} f_{R_i,K}]^T$$

- Vectors normalized, summed to RMAC descriptor





The 3D convolutional activation tensor is divided into overlapping square regions sampled at multiple scales to compute the RMAC descriptor which captures localized features.

CBIR for Handwitten Notes

Datasets for Image Retrieval

- Dataset 1 (”db 0”) contains 120 unprocessed note images, with 20 images per class, from six different subjects.
- Dataset 2 (”db 1”) consists of 120 images, divided into 60 without gridlines and 60 artificially generated images. No connections exist between these two classes.
- Dataset 3 (”db 2”) includes 200 images, comprising 20 without gridlines and 180 generated grid images using nine different styles. Associations exist between non-grid and generated grid images across ten classes, with 20 images per class.

CBIR for Handwitten Notes

Retrieval Performance Evaluation

- Average Precision (AP) measures relevance of top k retrievals to query

$$AP@k = \frac{1}{k}\sum_{i=1}^{k}\delta(y_i, y_q) \text{ where } \delta(y_i, y_q) = \begin{cases} 1 & \text{if } y_i = y_q \\ 0 & \text{otherwise} \end{cases}$$

- mAP calculates the mean AP across all queries

$$\text{mAP} = \frac{1}{N}\sum_{i=1}^{N} AP_i@k$$

- Results:
  - Pretrained CNNs achieved highest mAP
  - Demonstrates power of CNN features
  - UNet models performed worse, struggled on complex images
  - Trained UNet had lowest mAP due to overfitting

Table 4: mAP on db_0

Model	k = 3	k = 5	k = 10	k = 20
VGG16	0.997	0.997	0.993	0.951
VGG19	0.997	0.998	0.994	0.930
DenseNet	1.000	1.000	0.998	0.923
Trained UNet	0.983	0.975	0.864	0.680
Kaiming UNet 0	1.000	0.995	0.989	0.923
Kaiming UNet 1	1.000	1.000	0.998	0.945
Kaiming UNet 2	1.000	0.998	0.996	0.935

Table 5: mAP on db_1

Model	k = 3	k = 5	k = 10	k = 20
VGG16	0.972	0.955	0.927	0.894
VGG19	0.950	0.937	0.905	0.876
DenseNet	0.972	0.958	0.932	0.894
Trained UNet	0.908	0.858	0.806	0.758
Kaiming UNet 0	0.939	0.903	0.854	0.768
Kaiming UNet 1	0.939	0.912	0.864	0.785
Kaiming UNet 2	0.906	0.868	0.837	0.740

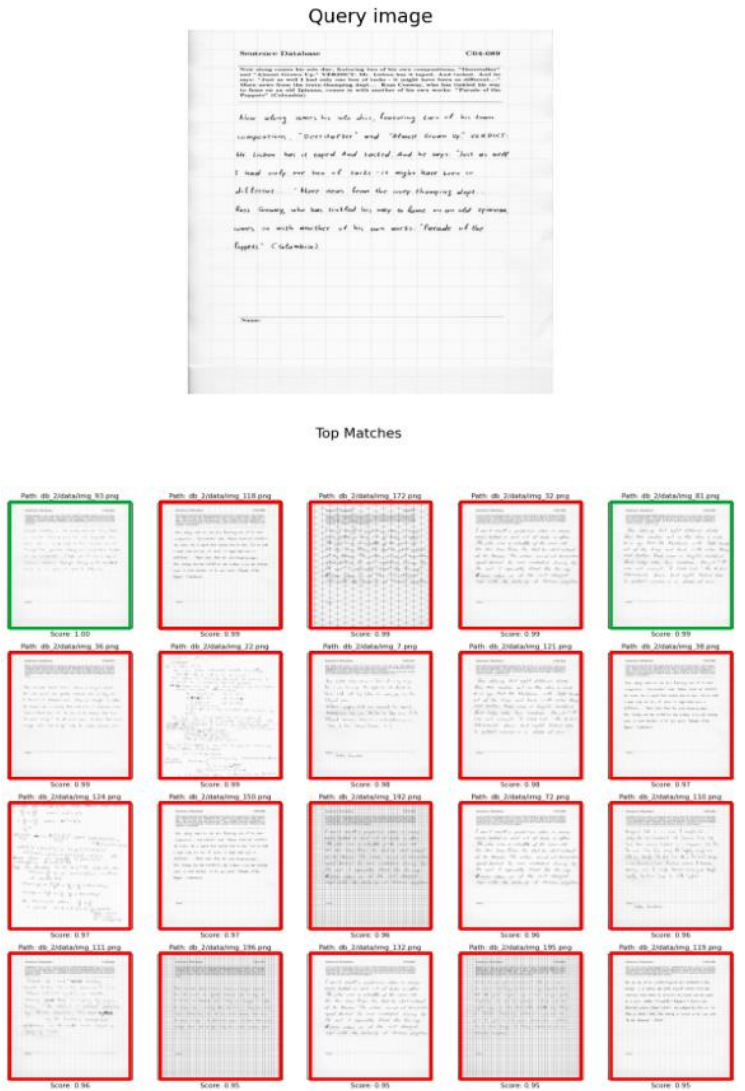
Table 6: mAP on db_2

Model	k = 3	k = 5	k = 10	k = 20
VGG16	0.675	0.636	0.604	0.548
VGG19	0.717	0.711	0.674	0.594
DenseNet	0.730	0.724	0.677	0.613
Trained UNet	0.467	0.360	0.317	0.302
Kaiming UNet 0	0.467	0.360	0.354	0.348
Kaiming UNet 1	0.467	0.370	0.371	0.375
Kaiming UNet 2	0.467	0.360	0.347	0.336

CBIR for Handwitten Notes

Some Tests

- *VGG16* on *db_0, k = 10*
- *VGG16* on *db_2, k = 20*



Conclusion

- UNet architecture excels in handwritten note image denoising, effectively removing gridlines and noise.
- The pipeline for generating synthetic training data and preprocessing real images shows promise for improving OCR and math symbol detection.
- UNet outperforms the Fourier transform method in gridline denoising.
- Data augmentation and the use of SGD optimization improve model training and performance.
- A retrieval system based on RMAC vectors can identify images requiring denoising.

Future Development

- The training dataset could be further augmented with more variations to better match real images.
- Instead of training UNet from scratch, fine tune a pretrained model could boost performance.
- Alternative learning rate schedulers like cyclical or one policies could be explored.
- More advanced metrics beyond MSE/SSIM would better evaluate similarity of denoised images to ground truth.
- Using Vision Transformer (ViT) or DINO for image embeddings may improve the retrieval system.
- Retrieval system could be extended to find similarity between crops rather than full images.

Thanks for the attention
cit. Transformer