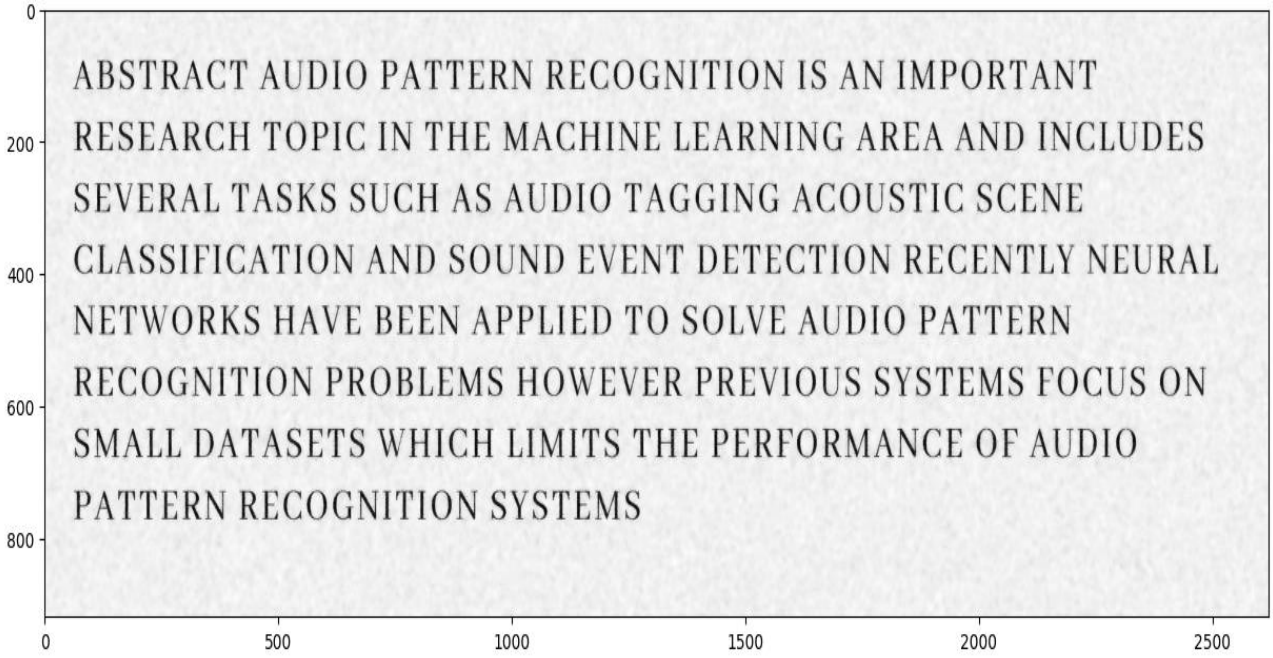
# Report on Project_5

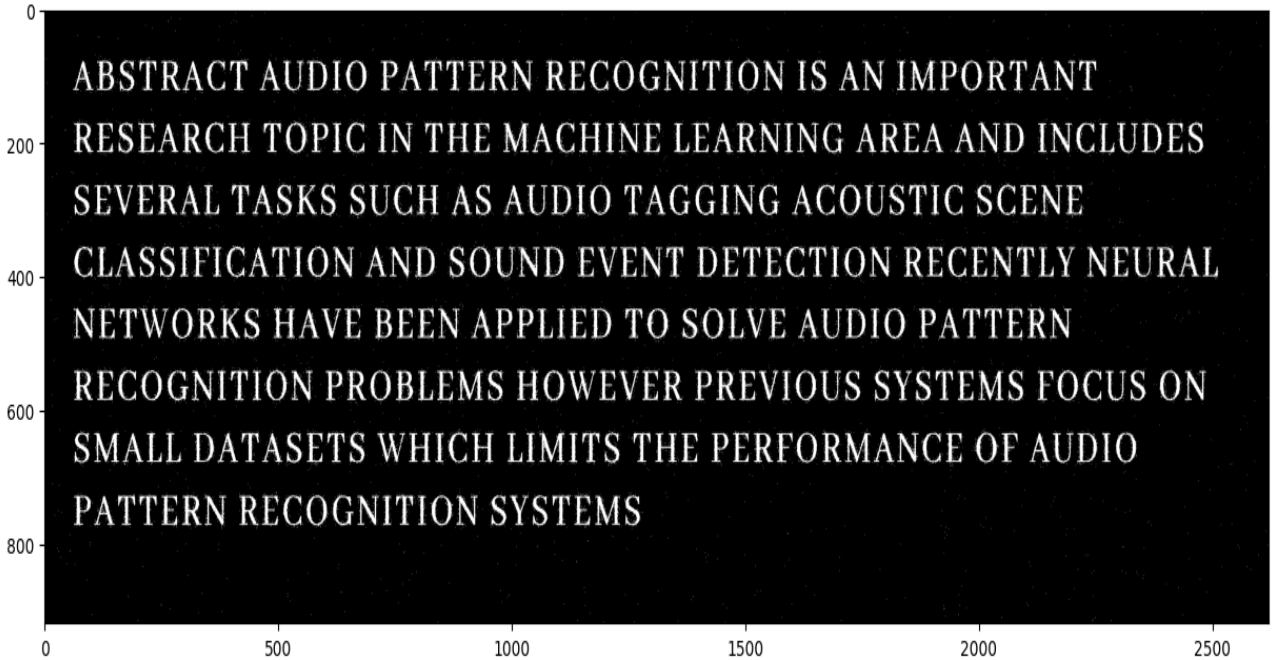
## Image Transcription

In this assignment the input was an image with texts and the output is save the texts into a file. I followed the following steps:

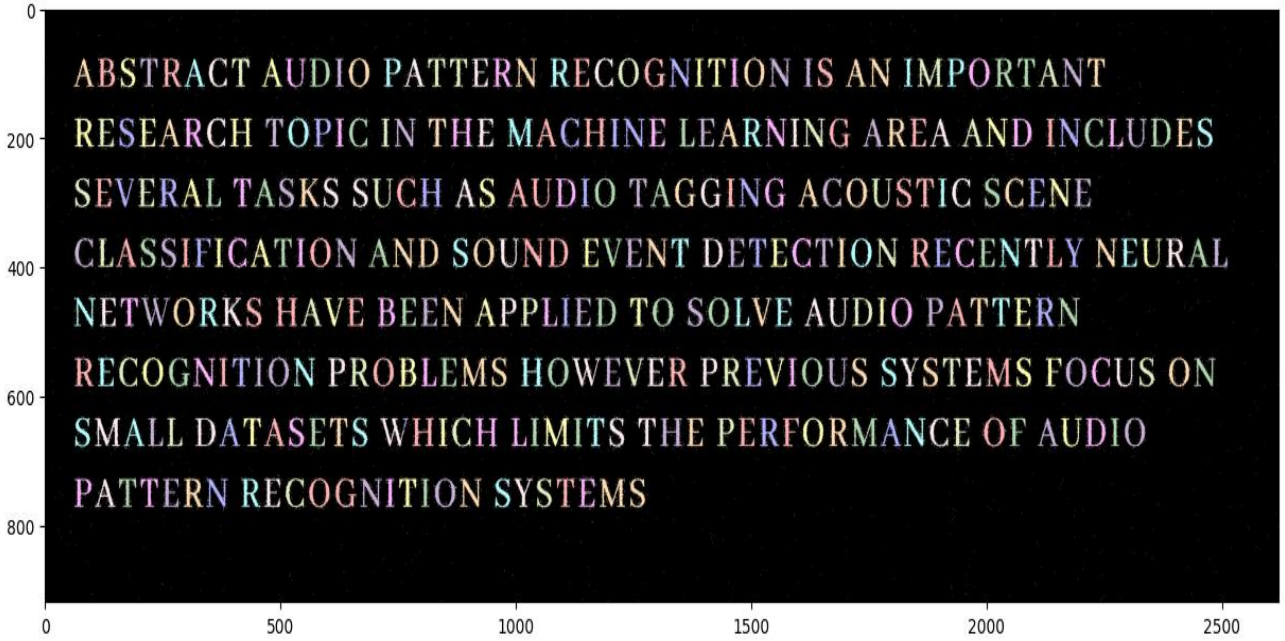
1. Denoise the image by using one of the non-filtering. I used bilateral filtering in this case.




2. Binarize the image by setting the pixel values to one for pixels larger than a threshold. (I used 0.75 as threshold)



3. Then I got the connected components of the binary image by using **skimage.measure.label**




4. After that we filtered the connected components. We only took the components with following properties:
    a. Convex area / area > 0.95
    b. area > 200
5. Then the filtered regions are sorted because they are not in order. I followed the following steps:
    a. First, I sorted the regions by the y_min coordinate of the bounding boxes.
    b. Replaced the similar y_min values with same y_min values.
    c. Each unique y_min values is in the same line of text. So we separated the regions with same y_min values to differentiate regions according to their lines.
    d. Each list of regions i.e., each line of text is sorted by the x_min of the bounding box values.
    e. Finally, each list of regions is concatenated again.
    f. Track the position of the ending of each list of regions to add line break in output file.
6. Then the regions are preprocessed to get the trained model prediction.
    a. Crop each region according to their bounding box.
    b. Pad the cropped images with zero by 10 pixels using **np.pad**
    c. Apply bilateral filtering to remove noises.
    d. Resize the padded images to (28, 28) using **skimage.transform.resize**
7. Get predictions for each preprocessed images and save them to a file. Then calculate the accuracy of the prediction by comparing the predictions with the ground truths.
8. I also set up an algorithm to check for spaces after each index.
    a. Calculated the difference between x_max and x_min value of two consecutive regions.
    b. The differences are sorted and the value which is greater than 90% of all the differences is selected as threshold.
    c. Then the index the difference values greater than the threshold value are considered as spaces.

Some sample outputs are given below:

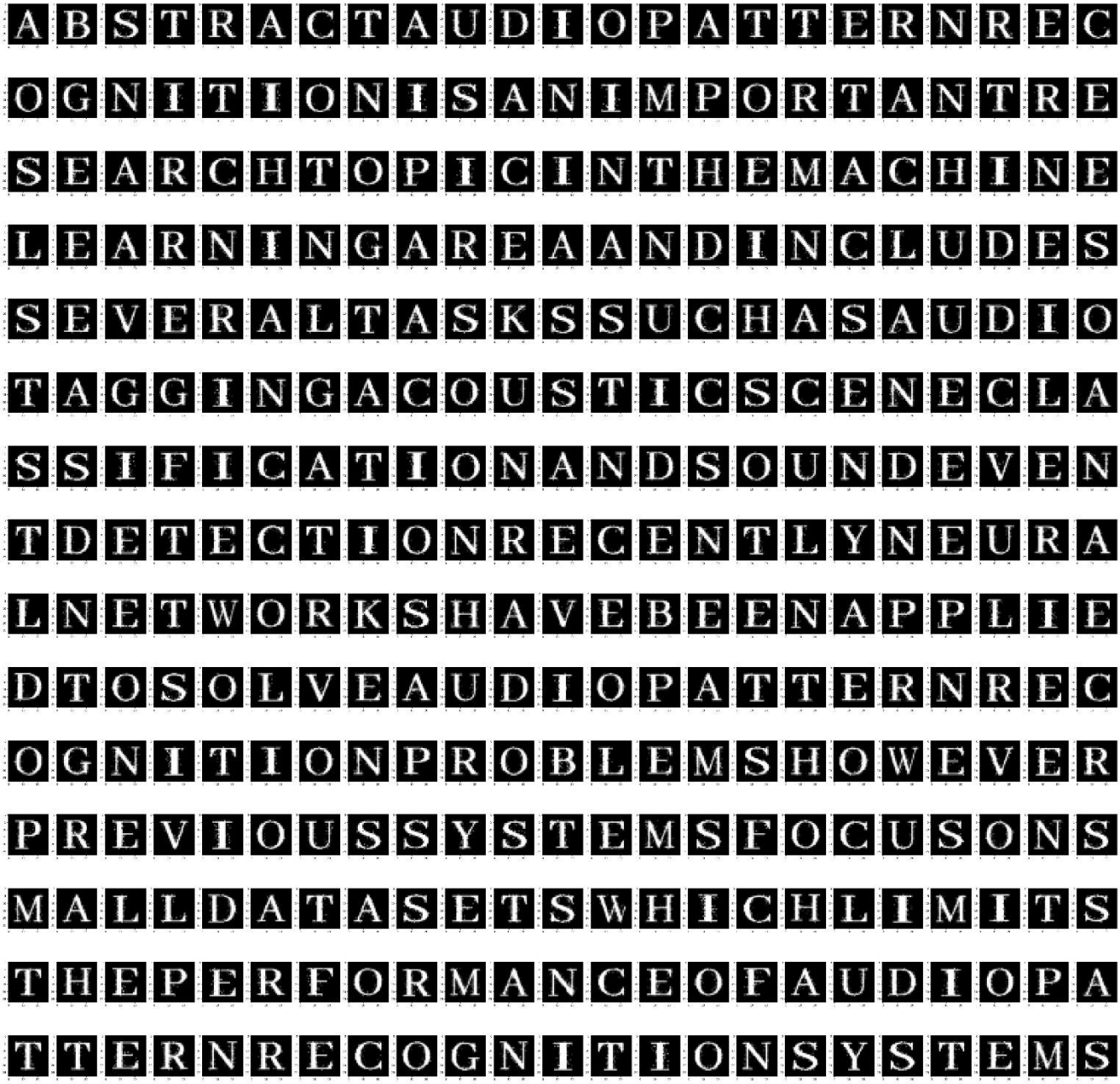



Figure: Each letter after filtering, sorting and preprocessing.

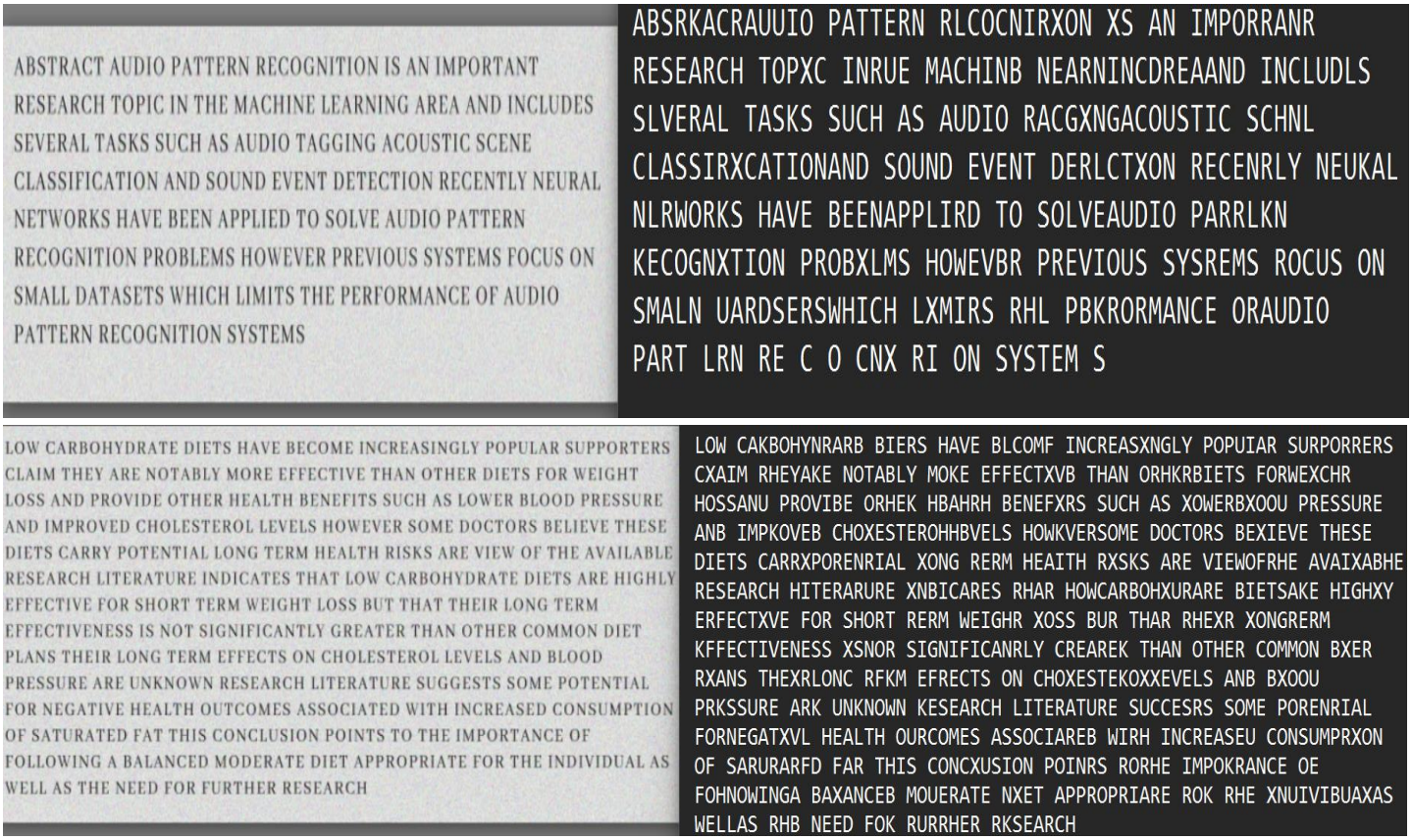



Figure: Left side is the input image. Right side is the models output written in a text file after running our full preprocessing pipeline.