Predict porosity from thin sections using DL

Problem statement:

Thin sections are prepared for quantitative textual analysis of the rock samples to understand the mineral composition, fabric, and general makeup of a rock. This technique is used for study of grain size, grain shape and roundness, porosity, and permeability of the reservoir rocks.

A thin sliver is cut from the rock sample with a diamond saw. Then they are cleaned in a vapor phase bath to remove solvable excess residual hydrocarbon. Next the samples are impregnated with blue epoxy to identify porosity and preserve textures, polished and mounted onto a glass slide. The samples are then ground down to a thickness of 30 microns and stained with a combined carbonate stain. Finally, a second glass slide is glued on the polished surfaces.

The prepared thin sections are examined petrographically. A polarizing microscope is used to take pictures of the thin section. When placed between two polarizing filters set at right angles to each other, the optical properties of the minerals in the thin section alter the color and intensity of the light as seen by the viewer. As different minerals have different optical properties, most rock forming minerals can be easily identified. Individual minerals are identified by their stained color and crystal structure (if visible).

To analyze a material's composition, the technician identifies rock constituents at several "points" (~300-400) within a single thin section. Petrographic analysis can be used to evaluate the pore system in a reservoir rock. These data are then used to calibrate 3d models of the subsurface and predict and estimate desired resources.

This project is aimed at developing a model that can predict porosity from the image of a thin section. Such a model can help to save on cost and human effort that is involved in thin section analysis.

Description of the dataset

I'm using data from Volvo dataset. It is a published dataset for people to use for their research in oil and gas industry. The dataset has 26 wells. Three out of twenty-six wells have core data and two of these three have thin section analyses data (thin section images and corresponding porosity estimate). The thin section image data is available as pdf reports and must be wrangled and extracted to be used in this study. Each thin section has two photographs of the same thin section. There is a depth reference for the rock sample used to create the images. It is shown on the picture in-between the two images partially covering each of them. Picture below shows a picture for a single thin section taken in two different modes.

A spread sheet for conventional core analyses reports rock porosity for a particular depth. We can relate two datasets through the common variable – depth.

Figure: An example of a thin section photograph from a rock sample in well 15/9-19-BT 2 taken at depth 4061,75 m MD RKB which has the following rock properties: porosity – 23.7%

Data wrangling

The first step is to extract thin section images from the pdf reports along with their associated depth references.

The image extraction from pdf with fitz package produces the result shown in Figure 2. We can see a pair of thin sections are show on the same image along with some background. The thin section photographs were presumably printed and placed in the sheet protector on some white background.

We first extract text from each of the images and search for depth related information using regular expressions. The process is somewhat successful, but still requires manual qc's and filling missing depths. We save depth references for both thin sections as part of the image name when saving the images on the disk.

The next step is to extract individual images from each page. There are two thin sections on each pdf page with two different photograths (left and right). I call them top left, top right, base left, and base right. The location of the separations lines between individual images were used to crop each photograph from the page. The coordinates for the cropping are founds by summing the values of the pixels along horizontal or vertical directions. White separation lines between the photographs will have lower sums than lines going through the photos. Searching within a particular region for min or max of the low sum clusters will give the coordinates of the photos. Example of the extraction of the image coordinates for a page shown in Figure 2 is found in Figure 3.

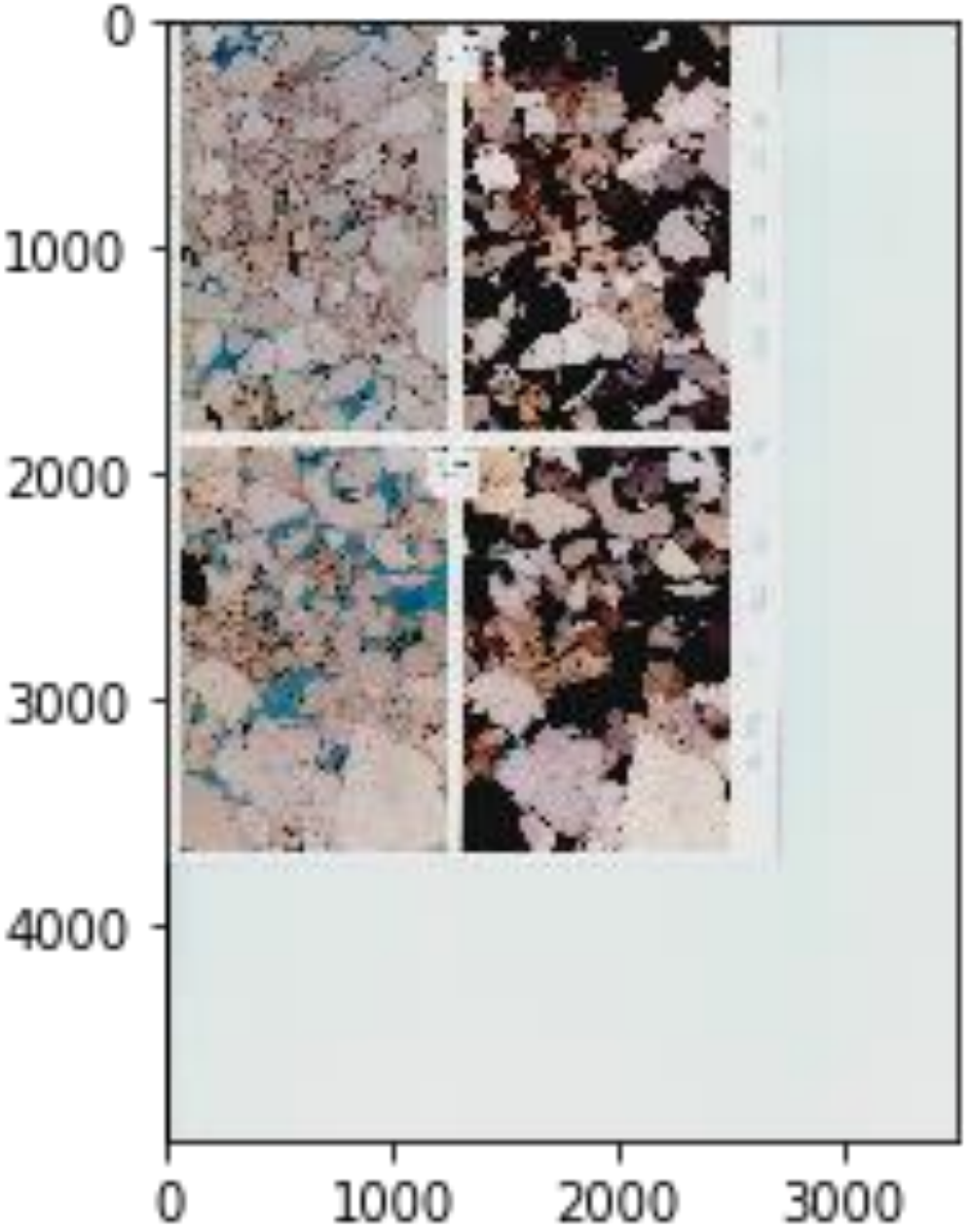


Figure 2: Image extraction from pdf report using fitz package and saved under name 15_9_19_A_p2-78-3837.55_3838.50. The name for this image is a combination of well name (15_9_19_A), page number (p2) in the pdf report, image id (78), and the depths shown for each thin section in the image (3837.55 for the top thin section, 3838.50 for the base thin section).

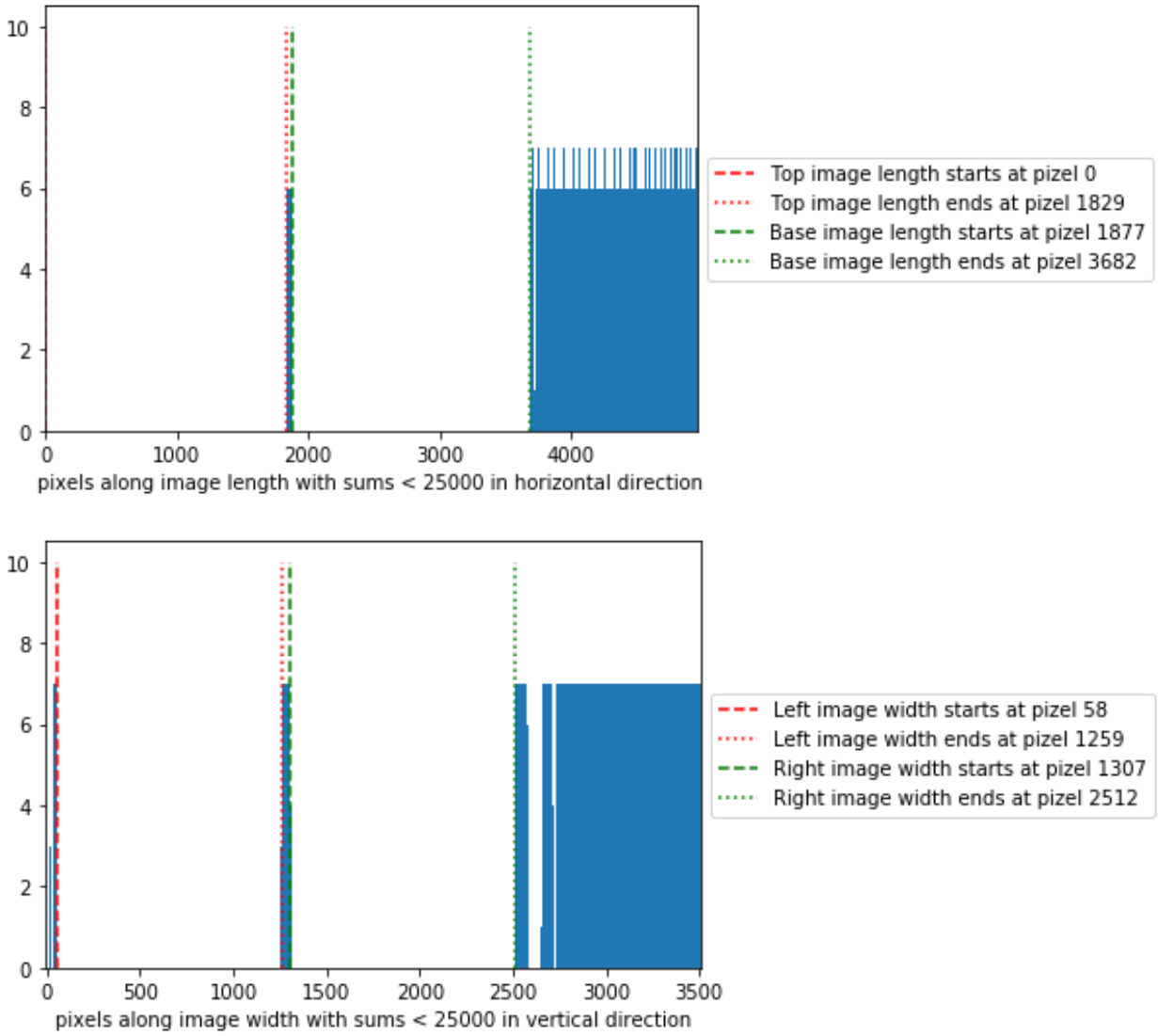


Figure 3: Vertical lines are showing the cropping coordinates for extracting four images from the Figure 2. Top figure is for extraction along the length of the page to separate top images from the base ones. The lower figure is for extraction along the width of the page to separate left images from right ones.

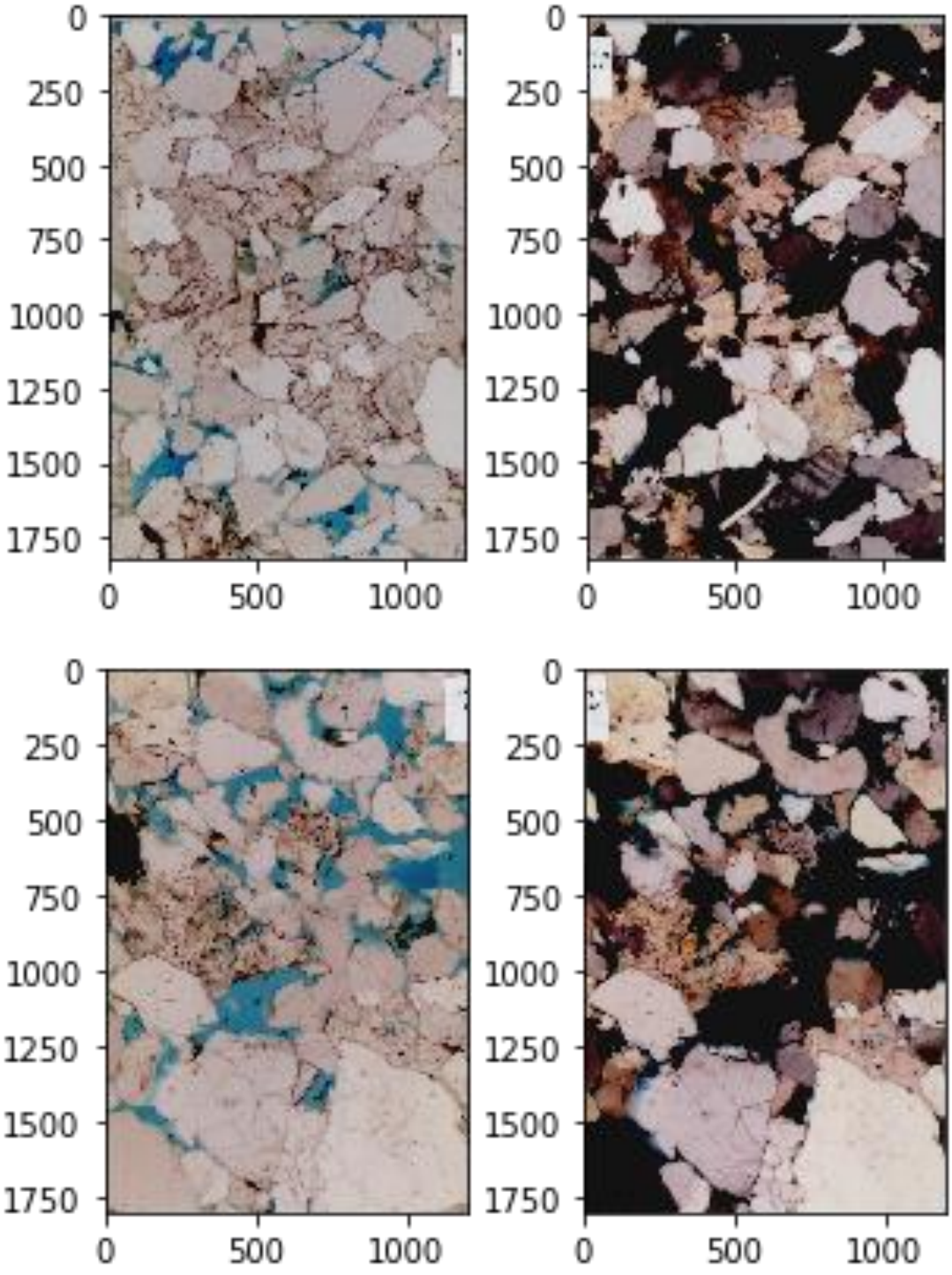


Figure 3: Example of the image extraction for images shown in Figure 2 (15_9_19_A_p2-78-3837.55_3838.50). Four images are extracted: top left and top right are the photographs of the same thin sections at depth 3837.55 m MD RKB; base left and base right are the photographs of the same thin sections at depth 3838.50 m MD RKB

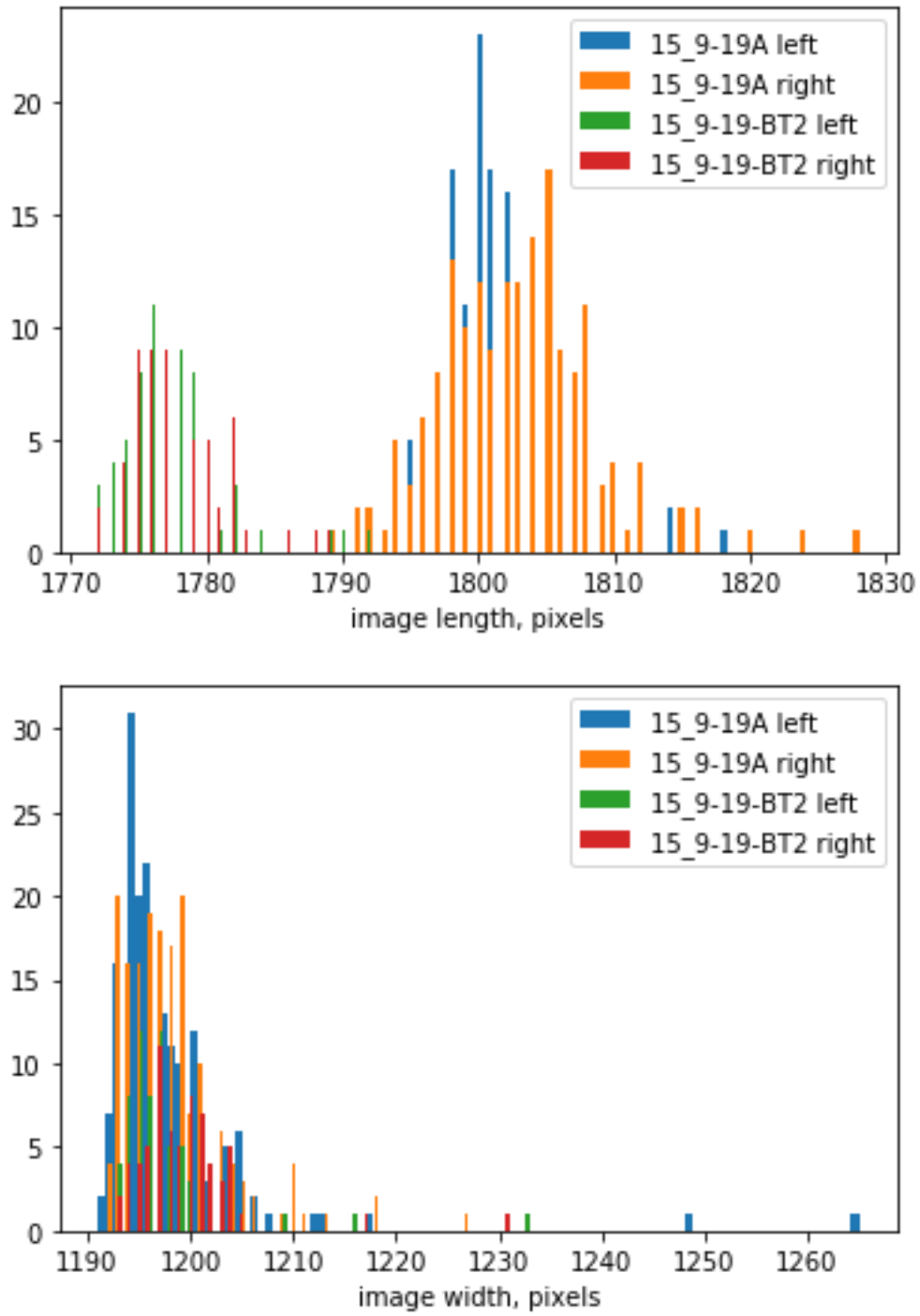


Figure 4. Histogram of image depths and widths for all images extracted for well 15_9-19A (blue) and well 15_9-19TB2 (orange).

The process of image extractions results with the images of different sizes. Figure 4 shows the histograms of image lengths and widths for all the images extracted from the reports. The widths of the images have a narrow distribution across two wells and both left and right photographs. The lengths

have a bimodal distribution. Both thin section photos from the 15_9-19TB2 are shorter than those from 15_9-19A wells.

In order to use all the data in our analyses images were resized to the standard shape of 1024 by 1536.

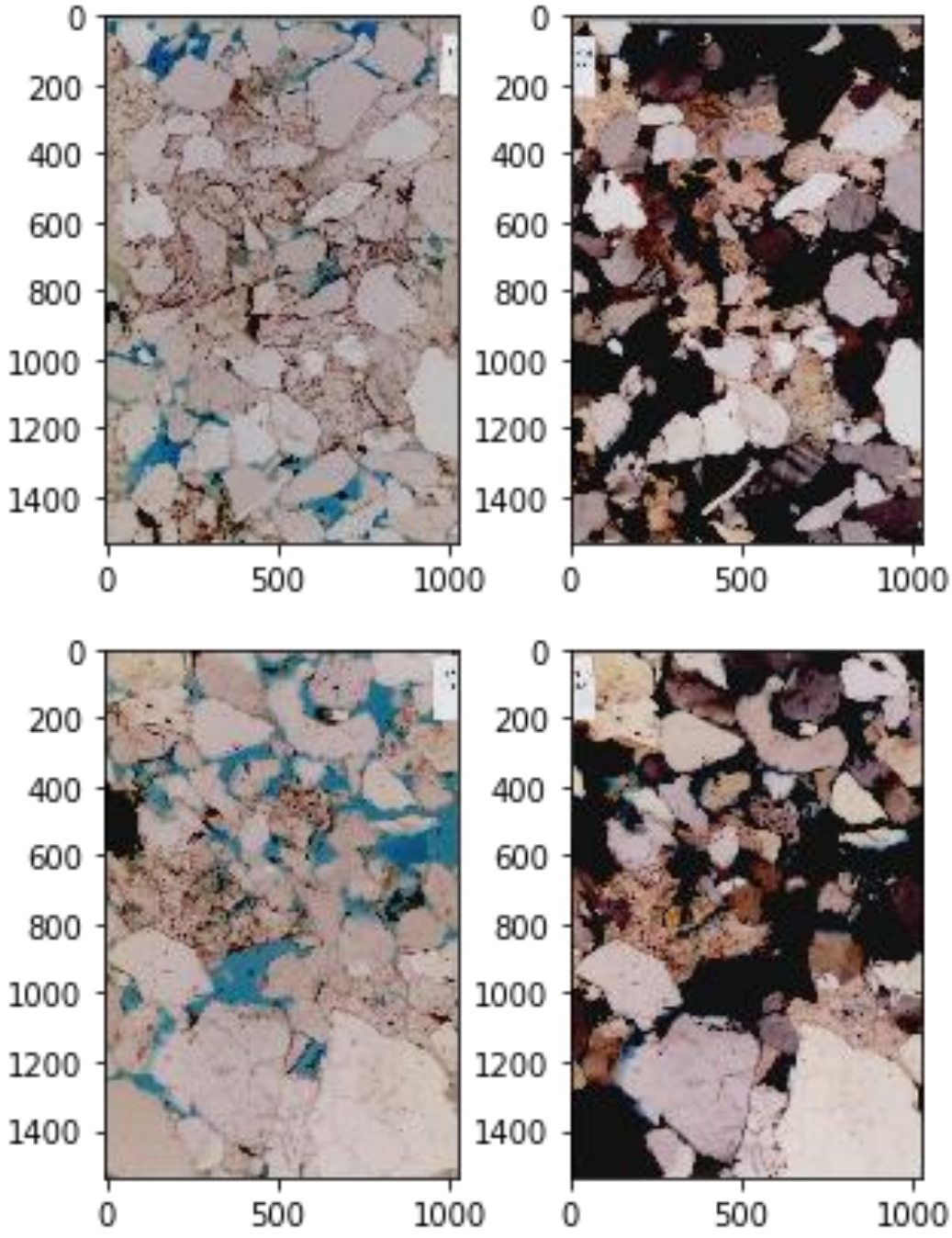


Figure 5: Example of the image resizing to the standard shape of 1024 by 1536 for images shown in Figure 3 (15_9_19_A_p2-78-3837.55_3838.50).

The final step in the data preparation is to find image labels (porosity values). As mentioned above porosities are reported in the excel spreadsheets for each depth. Porosity and corresponding depths are extracted from the core analyses spreadsheet into a data frame. Filenames are then added as a separate column by depth key. The resulting data frame contains porosity, depth values and filename for the thin section images.

	Depth, m	Porosity, pc	File Name
0	3837.55	10.8	15_9_19_A\15_9_19_A_p2_3837.55.png
1	3838.50	17.2	15_9_19_A\15_9_19_A_p2_3838.50.png
2	3839.40	12.7	15_9_19_A\15_9_19_A_p3_3839.40.png
3	3840.45	21.0	15_9_19_A\15_9_19_A_p3_3840.45.png
4	3841.45	22.1	15_9_19_A\15_9_19_A_p4_3841.45.png

Figure 6. A snapshot of a final dataframe with all the information required for training a DL model. Labels are stored in “Porosity, pc” column. Input are images saves under file names in the “File Name” column.

Data Preprocessing and Augmentation

For data preprocessing standard scaling approach was used. Three channel rgb image was used for training. Images were normalized from 0 to 1 by dividing the data by 255.

Image augmentation was performed by flipping each image in three different ways: along horizontal direction, along vertical direction, along horizontal and then vertical direction. The Figure 6 below shows the result of image augmentation.

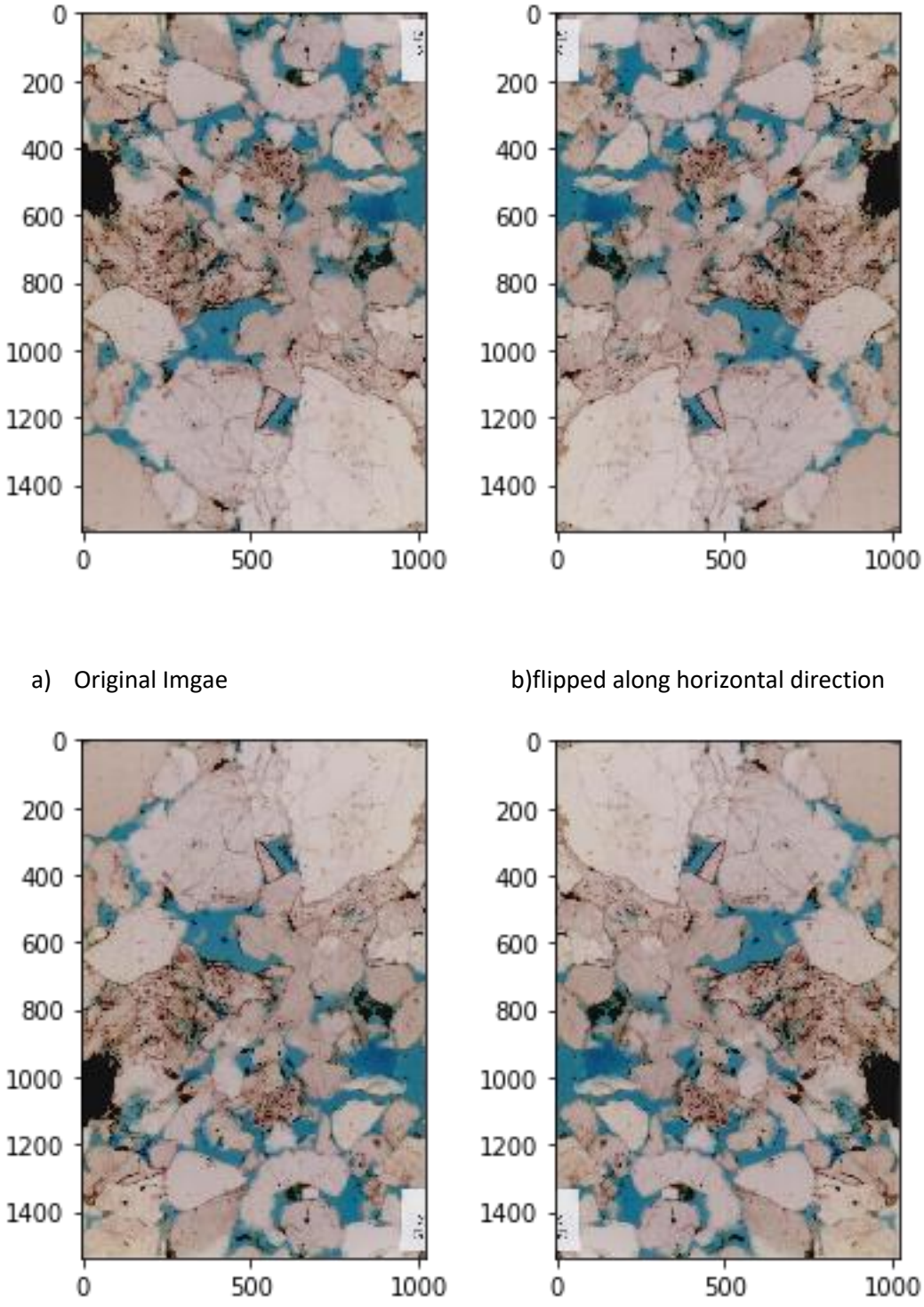
Well A has 152 thin section before data augmentation and 608 images after.

Well B has 66 thin section before data augmentation and 164 images after.

Data is split in training, validation, and testing datasets.

For the first experiment we use data from well A to train a model. Well A images are randomly split into training and validation sets using a ration of 90:10. Well B data is used for testing model generalization.

For the second experiment we combine well A and well B data and train the model on the combined dataset.



a) Original Imgae

b)flipped along horizontal direction

c)flipped along vertical direction

d) flipped along horizontal and vertical direction

Figure 6. Image Augmentation by flipping an original image (a) along horizontal direction (b), vertical direction (c), and horizontal and vertical direction (d)

Model Definition

CNN model was used to predict porosity values for thin section images. See Figure 7 for model detail. Two identical convolution models are trained on left and right images. The resulting weights are combined before flattening the output into the series of dense layers.

The original image size of 1536 by 1024 was used for the input. The images are then send through a serious of encoder blocks defined by 2d convolution layers with strides of 2 followed by max pooling, and batch normalization. Relu activation is used at the end of a block.

There is a total of 4 blocks in each CNN branch with the increasing number of filters: 64, 128, 256, 256). The first block has a stride of 3 and 2 and no batch normalization.

After combining the weights of two branches another 2d convolution operation with relu activation is performed.

The results are then flattened into a 128-neuron dense layer. Activation and batch normalization are performed. Dropout of 0.25 was used for model regularization.

This operation is repeated again with the dense layer pf 64, relu activation, batch normalization and dropout.

The last dense layer has one neuron and sigmoid activation.

The model has 1,656,193 total parameters with 1,653,249 of them trainable and 2,944 not-trainable.

Model Training on a single well

I used Adam optimizer with a learning rate scheduler and beta parameter of 0.5 for model training. The learning rate scheduler decays the learning rate with a cosine annealing for each batch as follows in Figure 8. Over 10 epochs the learning rate is decayed from 0.001 to 0.000001 and then restarts.

The learning rate scheduler help to make sure the learning in not stacked around the local minimum and converges to global optimum.

For the loss function I use mean absolute percentage error. I trained over 200 epochs with a batch size of 32. Early stopping was employed with a patience of 50 to avoid overfitting.

Figure 9. shows training and validation loss decreasing as a function of epoch. The best model was achieved at epoch 70 and has a training loss of 19% and validation loss of 21%. After epoch 70 the model starts to overfit as the validation loss is starting to increase.

Figure 10. show the cross plot of the actual porosity vs the predicted one for the validation dataset (10% hold-off data from well A). The majority of points are sitting around a forty-five-degree line, which suggests reasonable performance.

Figure 11. show the cross plot of the actual porosity vs the predicted one for a testing dataset (well 15/9-19-BT 2 data that was not used in training). The prediction for well 15/9-19-BT 2 is not very good. In most of the cases true porosity is overpredicted. Th model did not generalize well onto the data from the second well. There can be a lot of reasons for that including label subjectivity, different image color

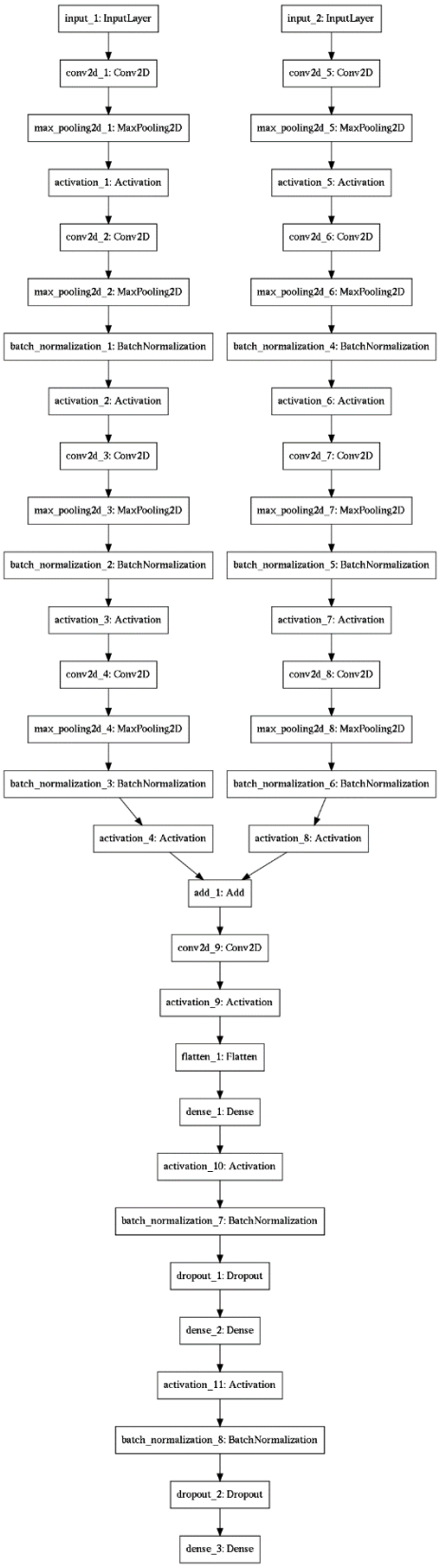

input_1: InputLayer
input_2: InputLayer
conv2d_1: Conv2D
conv2d_5: Conv2D
max_pooling2d_1: MaxPooling2D
max_pooling2d_5: MaxPooling2D
activation_1: Activation
activation_5: Activation
conv2d_2: Conv2D
conv2d_6: Conv2D
max_pooling2d_2: MaxPooling2D
max_pooling2d_6: MaxPooling2D
batch_normalization_1: BatchNormalization
batch_normalization_4: BatchNormalization
activation_2: Activation
activation_6: Activation
conv2d_3: Conv2D
conv2d_7: Conv2D
max_pooling2d_3: MaxPooling2D
max_pooling2d_7: MaxPooling2D
batch_normalization_2: BatchNormalization
batch_normalization_5: BatchNormalization
activation_3: Activation
activation_7: Activation
conv2d_4: Conv2D
conv2d_8: Conv2D
max_pooling2d_4: MaxPooling2D
max_pooling2d_8: MaxPooling2D
batch_normalization_3: BatchNormalization
batch_normalization_6: BatchNormalization
activation_4: Activation
activation_8: Activation
add_1: Add
conv2d_9: Conv2D
activation_9: Activation
flatten_1: Flatten
dense_1: Dense
activation_10: Activation
batch_normalization_7: BatchNormalization
dropout_1: Dropout
dense_2: Dense
activation_11: Activation
batch_normalization_8: BatchNormalization
dropout_2: Dropout
dense_3: Dense

Figure 7. CNN model architecture for training with both left and right images.

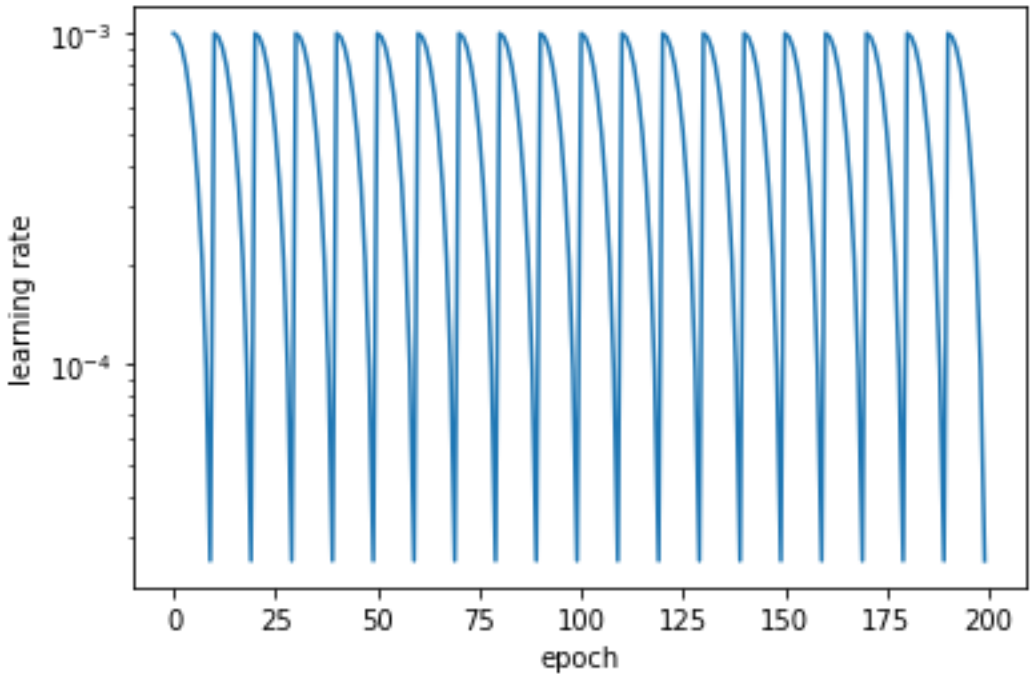


Figure 8. The learning rate scheduler with a cosine annealing. Decay learning rate from 0.001 to 0.000001 over 10 epochs and restarts.

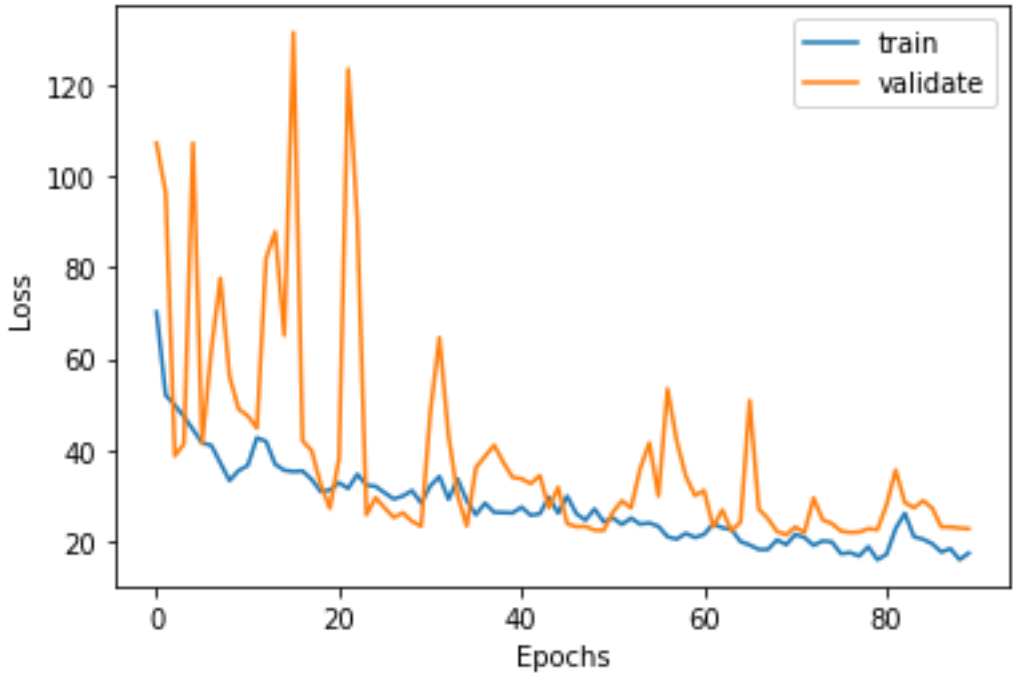


Figure 9. Change in loss function as a function of training epoch for the training (blue) and validation (orange) sets. Only well 15/9-19-A dataset is used for training and validation.

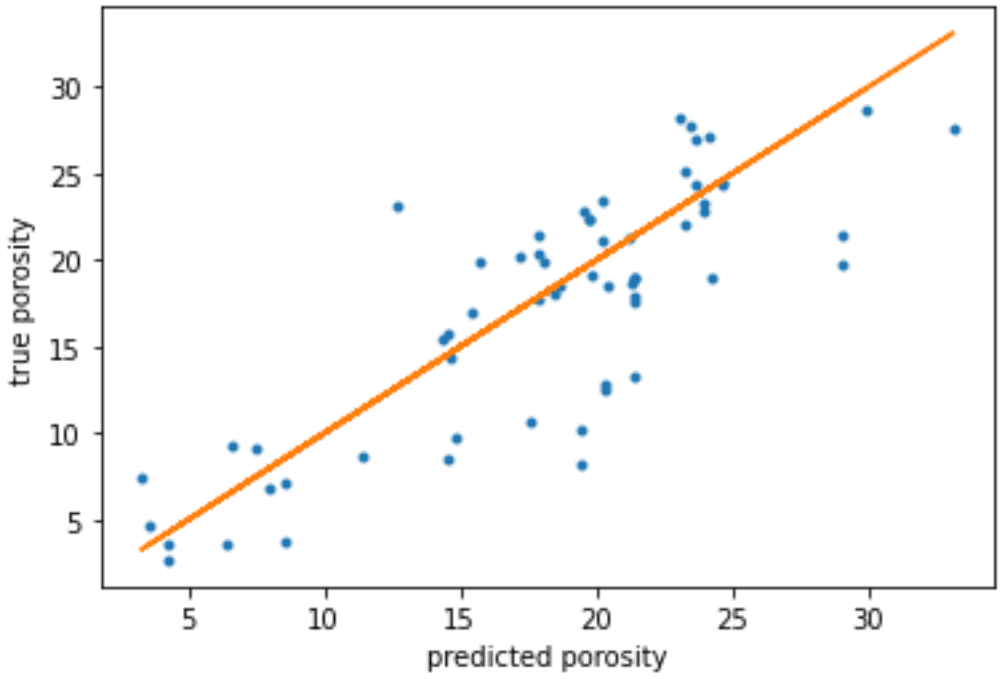


Figure 10. The cross plot of the actual porosity vs the predicted one for the validation set (blue dots). Forty-five-degree line for perfect prediction is shown in orange. Coefficient of determination for the linear fit is 0.674.

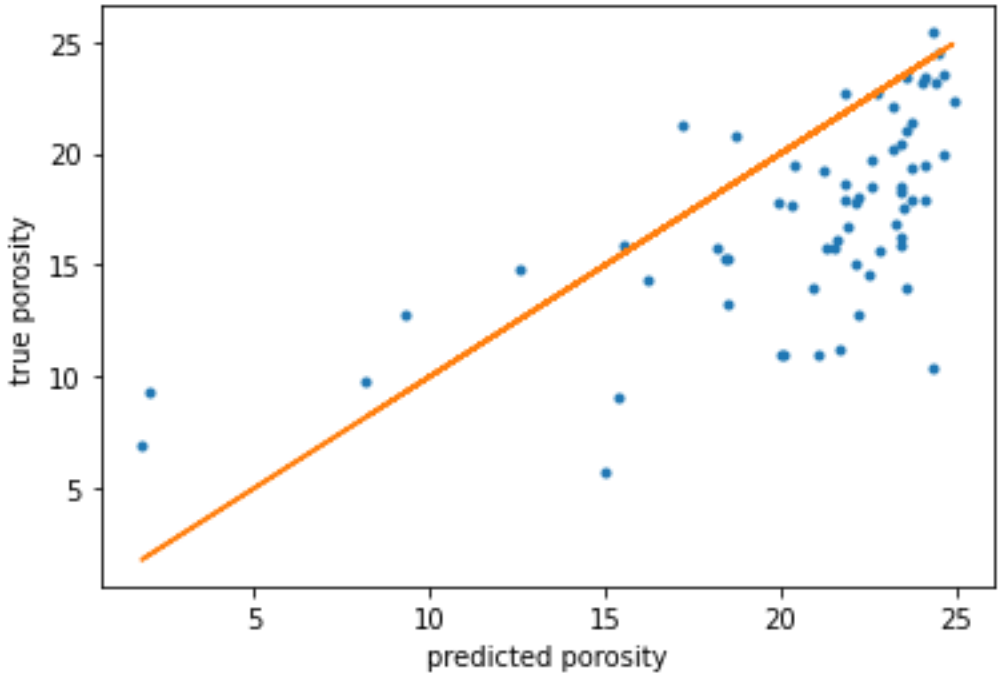


Figure 11. The cross plot of the actual porosity vs the predicted one for the testing set (well 15/9-19-BT 2). Forty-five-degree line for perfect prediction is shown in orange. Coefficient of determination for the li near fit is 0.391.

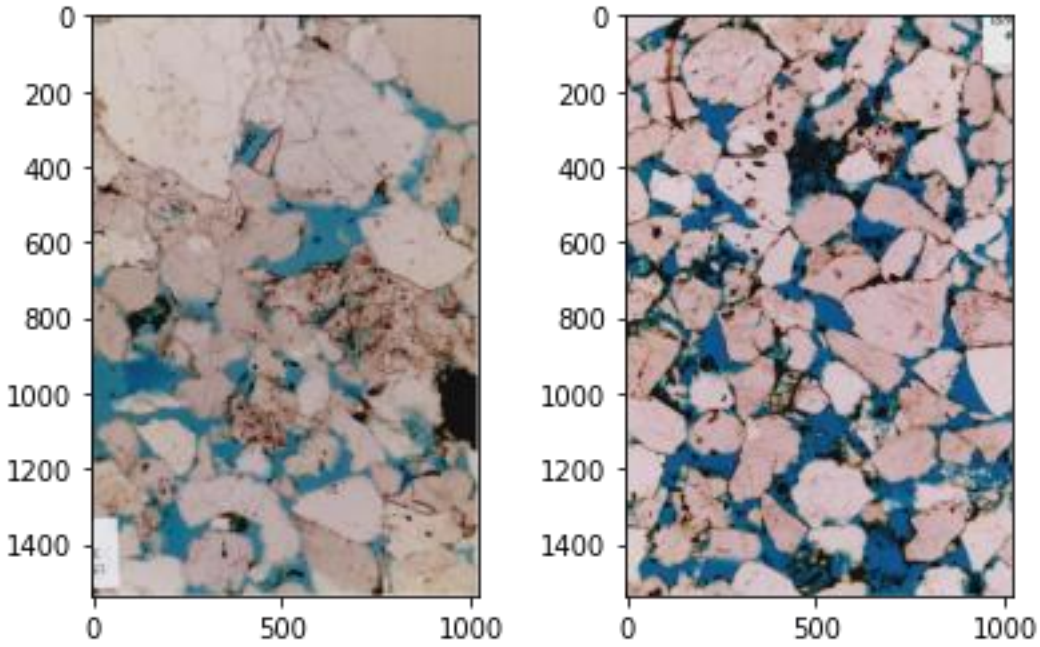


Figure 12. Difference in color palettes from images in well 15/9-19 A (left) and images in well 15/9-19-BT 2 (right). The blue color on the right image is much deeper that blue color on the left image. Blue color correspond to porosity value.

pallets (blue color in images from well 15/9-19-BT 2 is deeper than blue color in images from well A, see Figure 12).

Model Training on a both wells

For the next experiment I trained a model using both wells. The same 90:10 ratio was used for training-validation split. The learning curve is shown in Figure. 13. The best model was achieved at epoch 70 and has a training loss of 19% and validation loss of 25%. After epoch 41 the model starts to overfit as the validation loss is starting to increase.

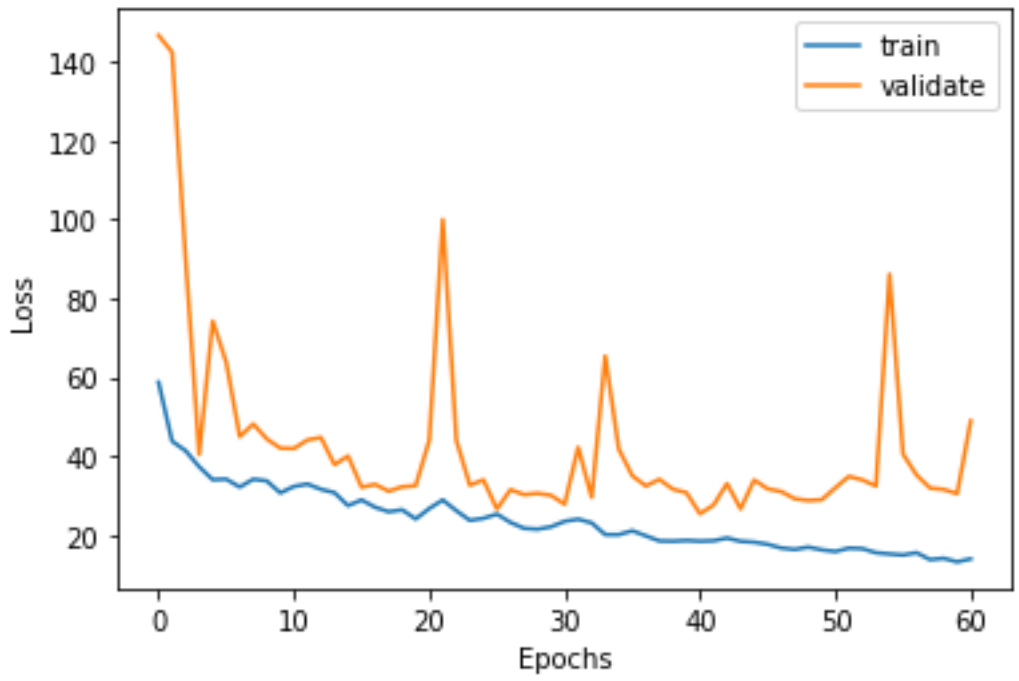


Figure 13. Change in loss function as a function of training epoch for the training (blue) and validation (orange) sets for combined dataset.

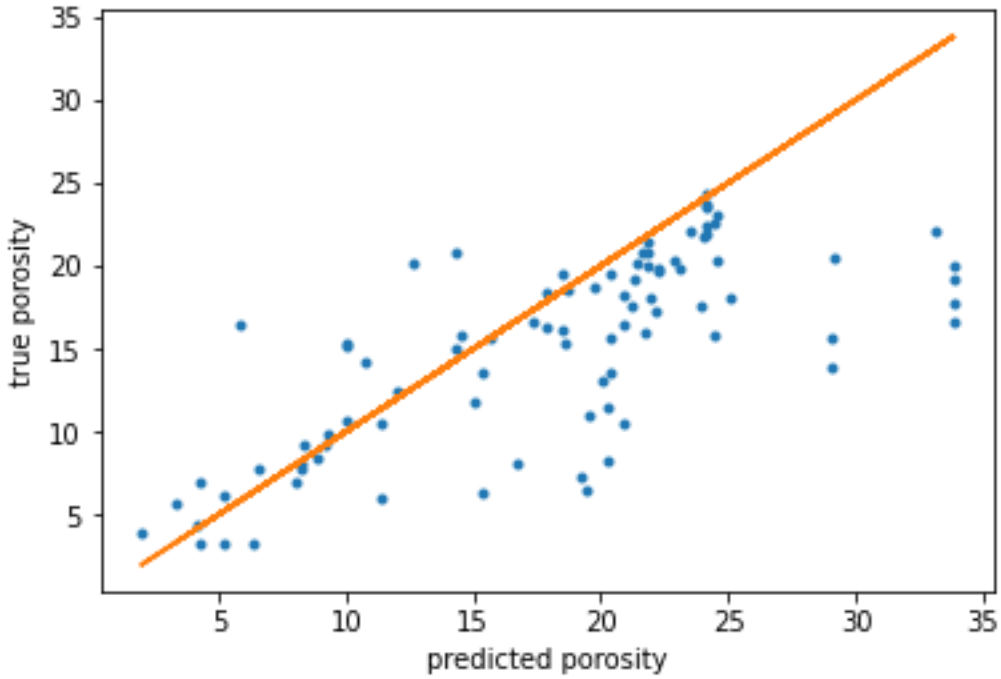


Figure 14. The cross plot of the actual porosity vs the predicted one for the testing set (both wells). Forty -five-degree line for perfect prediction is shown in orange. Coefficient of determination for the linear fit is 0.530.

Discussion

Although I have not achieved perfect results predicting porosity from the images of thin section, I have made a good progress so far. There are still some images that I overpredict the porosity for. This issue needs to be investigated further by error analyses.

The datasets I'm working with are not ideal. These are images cropped from the pdf files. I believe they will benefit for more careful preprocessing workflow and normalization procedures.

As the next step I could also tune model hyperparameters even further or even experiment with different model architectures.

Some form of additional data augmentation could be useful for this problem. From Figure 15. It is clear that we do not have enough data for low and high porosity samples and the bulk of the images have porosity from 20 to 25 %. Model poor prediction at the highest porosity end can be explained by this label disbalance.

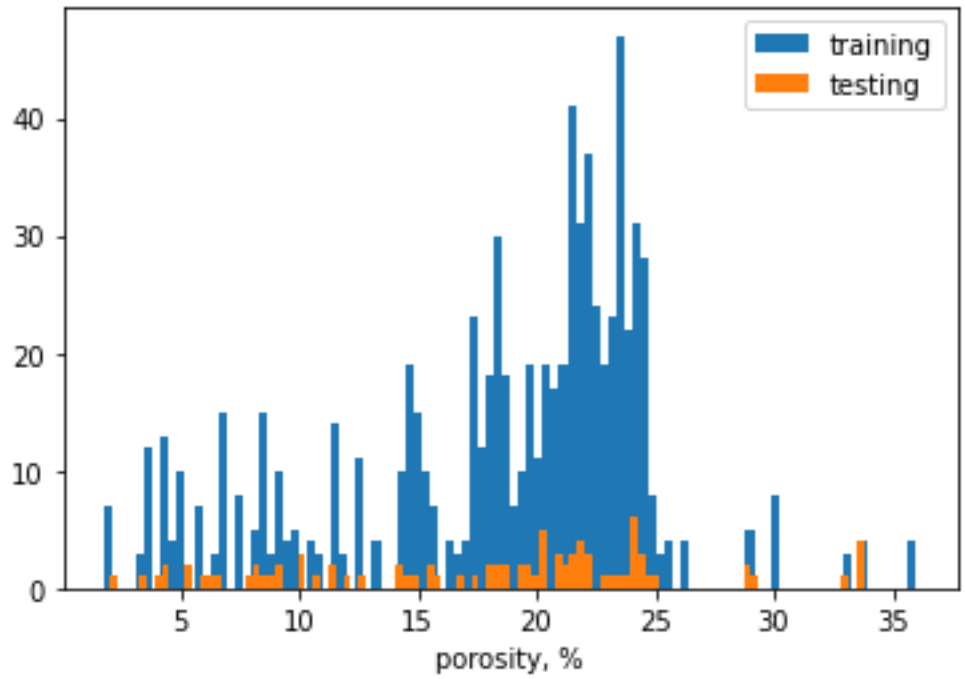


Figure 15. Distribution of porosity in the training (blue) and the testing (orange) datasets. Disbalance in the proportion of data between low, mid, and high porosity ranges. There is not enough data for the low and high porosity.