Multi Class Texture Analysis in Colorectal Cancer Histology Project

Noam Koren 208276907
Linoy Valko 206842742

DataSet

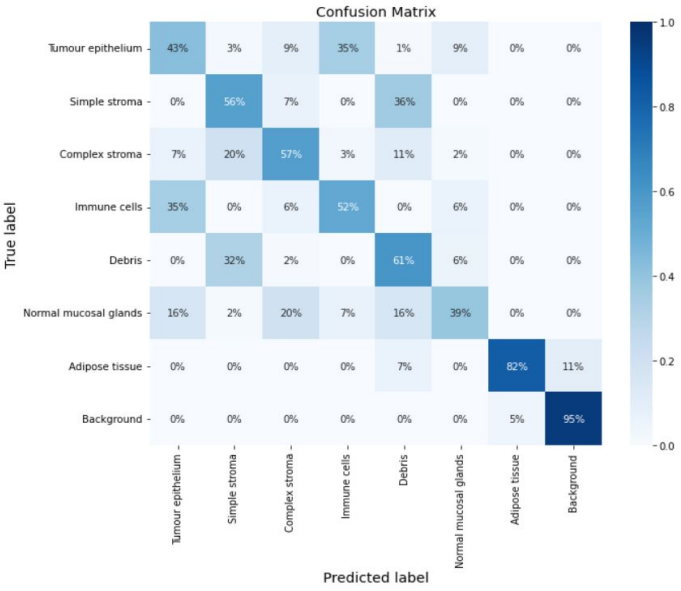
- The data we received compose of 5000 histological **images of human colorectal cancer** including **8 different types** of tissues: 'Tumour epithelium', 'Simple stroma', 'Complex stroma', 'Immune cells', 'Debris', 'Normal mucosal glands', 'Adipose tissue', 'Background'. The dimension of every picture was (150, 150, 3).
- This data was chosen specifically to train a wide range of **texture description and classifier**. As a result we found the **optimal classification method** from 9 different models.
- In the pre processing level, we divided the data in 255, because every pixel's range is between 0-255. By that, we convert int data set into **float data set** between 0-1.
- In the train- test split, We divided the data set into **90% train**, which is 4500 pictures, and **10% test**, which is 500 pictures.

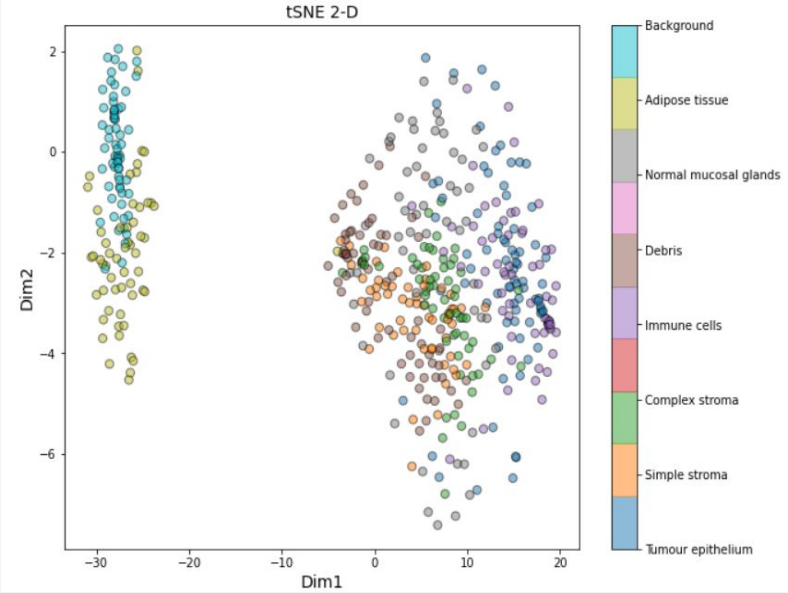
Experiment 1

Training accuracy: 99.6%
Validation accuracy: 60.6%
(*significant overfitting)

$\mathcal{X}i$ → Image Downsample →  → Reshape (Flatten) → $Zi \in \mathbb{R}^{3072}$ → SVM classifier → $yi \in \{1,2,3, \dots, 8\}$

(150, 150, 3) **(32, 32, 3)** **(3072,)**





- Dimensionality reduction: downsampling image to 32x32 and then flattening, resulting in a 3072-D feature space.
- Classifier: SVM classifier (kernel= linear, C= 1)
- tSNE: reduction from 3072-D to 2-D.

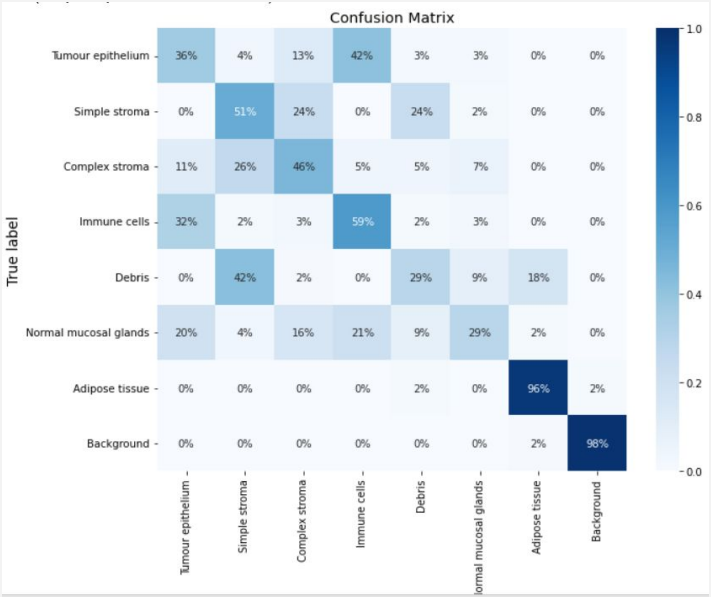
Experiment 2

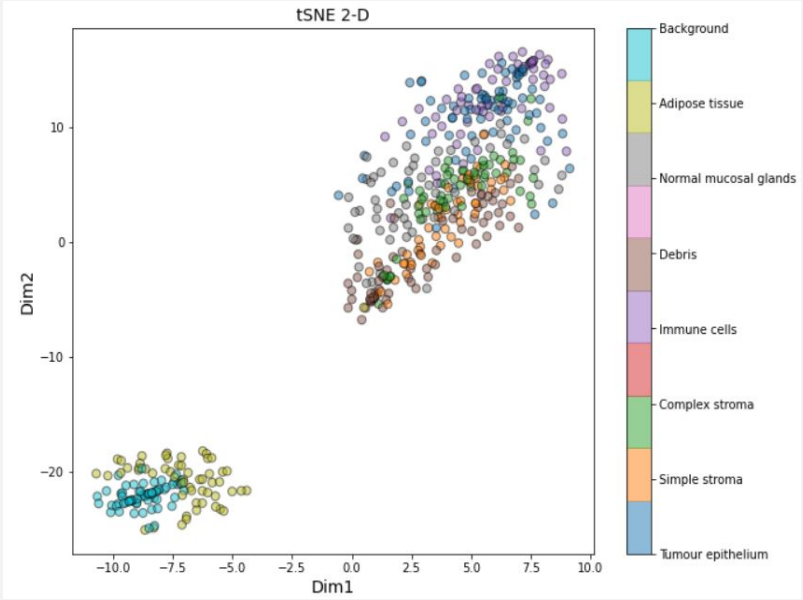
Training accuracy: 70.8%
Validation accuracy: 55%
(*significant overfitting)

x_i → Image Downsample →

→ Reshape (Flatten) → $z_i \in \mathbb{R}^{3072}$ → Softmax Classifier → argmax → $y_i \in \{1,2,3, \dots, 8\}$

(150, 150, 3) **(32, 32, 3)** **(3072,)**





• <u>Dimensionality reduction:</u> downsampling image to 32x32 and then flattening, resulting in a 3072-D feature space.

• <u>Classifier:</u> Softmax classifier (trained for 30 epochs, verbose= 1, fully connected).

• <u>tSNE:</u> reduction from 3072-D to 2-D

Experiment 3

Training accuracy: 61.6%
Validation accuracy: 50.8%
(*overfitting)

xi

(150, 150, 3)

Image Downsample

(32, 32, 3)

Reshape (Flatten)

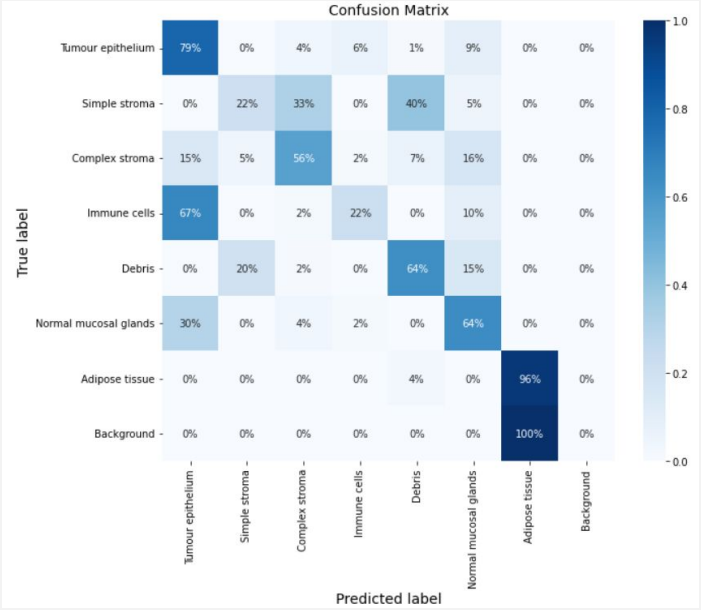
$Zi \in \mathbb{R}^{3072}$

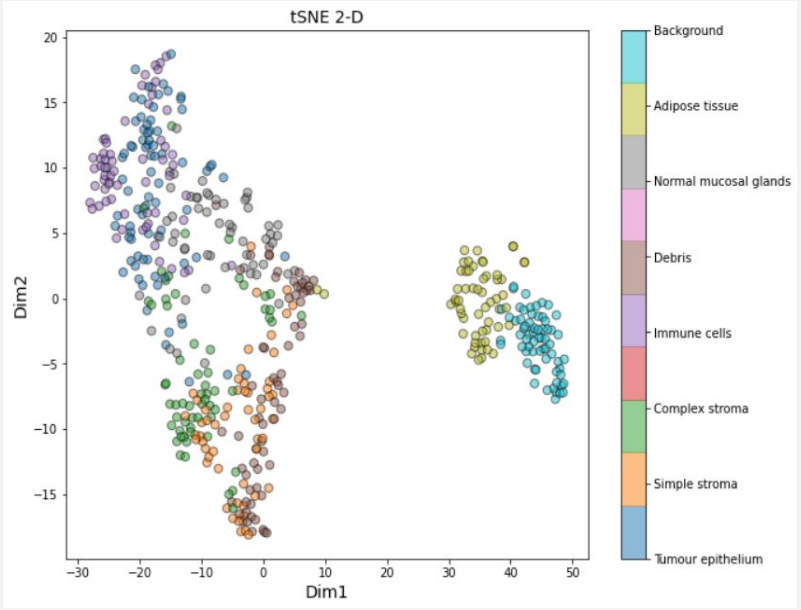
(3072,)

Neural Network Classifier

argmax

$yi \in \{1,2,3, \dots , 8\}$





• Dimensionality reduction: downsampling image to 32x32 and then flattening, resulting in a 3072-D feature space.

• Classifier: Neural Network dimension reduction to 128-D by ‘relu’ (fully connected), and Classification by ‘softmax’ (trained for 30 epochs, verbose= 1, fully connected).

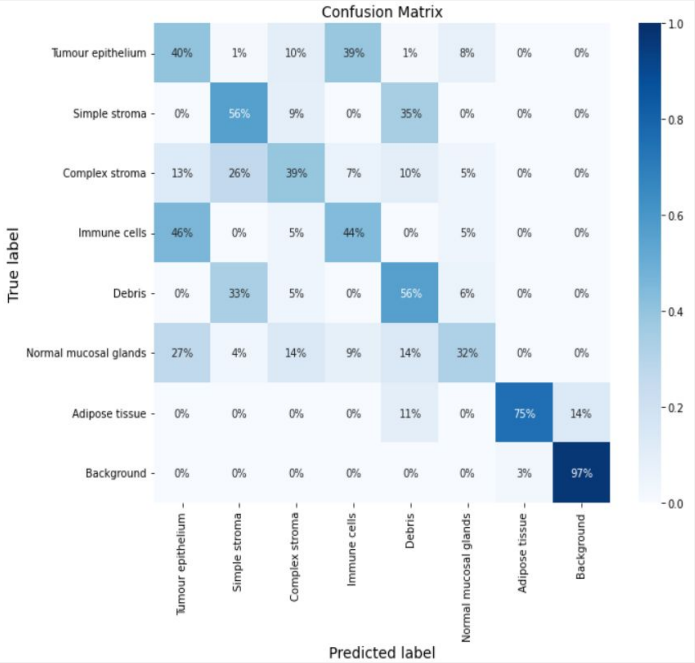
• tSNE: reduction from 128-D to 2-D.

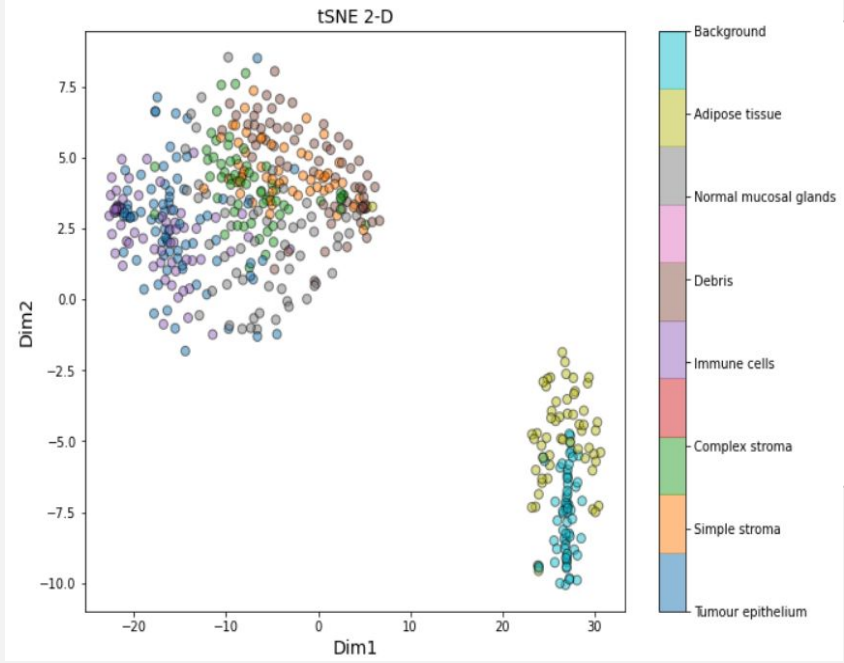
Experiment 4

Training accuracy: 88.8%
Validation accuracy: 55%
(*significant overfitting)

$\mathcal{X}i$ → Reshape (Flatten) → $Zi \in \mathbb{R}^{67500}$ → PCA → $Zi \in \mathbb{R}^{256}$ → SVM Classifier → $yi \in \{1,2,3, \dots, 8\}$

(150, 150, 3) **(67500,)** **(256,)**





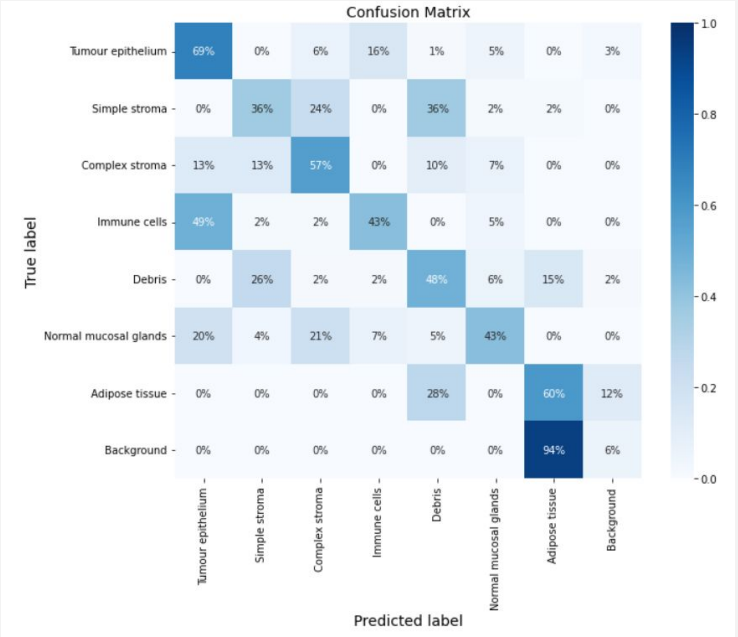
- Dimensionality reduction: reshape to 67,500-D, then PCA, resulting in a 256-D feature space.
- Classifier: SVM classifier (kernel= 'linear', C = 1)
- tSNE: reduction from 256-D to 2-D.

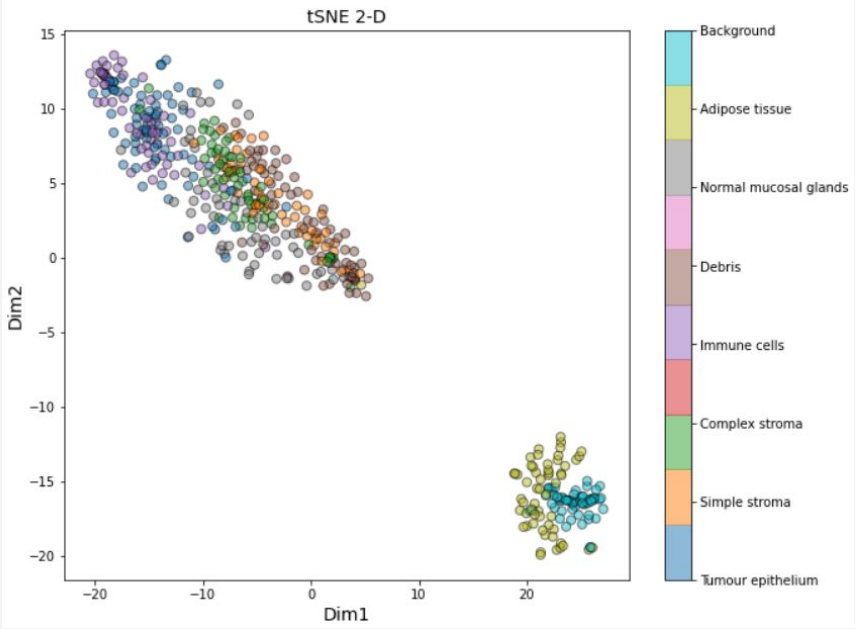
Experiment 5

Training accuracy: 58.9%
Validation accuracy: 45.8%
(*overfitting)

xi → Reshape (Flatten) → $Zi \in \mathbb{R}^{67500}$ → PCA → $Zi \in \mathbb{R}^{256}$ → Softmax Classifier → argmax → $yi \in \{1,2, \dots , 8\}$

(150, 150, 3) **(67500,)** **(256,)**





- Dimensionality reduction: reshape to 67,500-D, then PCA, resulting in a 256-D feature space.
- Classifier: Softmax Classifier (trained for epochs=30, verbose=1, fully connected)
- tSNE: reduction from 256-D to 2-D.

Experiment 6

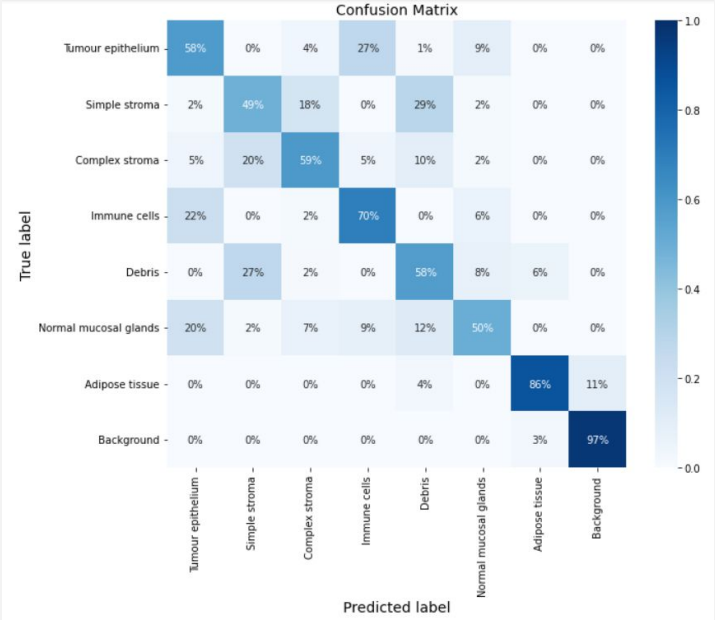
Training accuracy: 96.9%
Validation accuracy: 66%
(*significant overfitting)

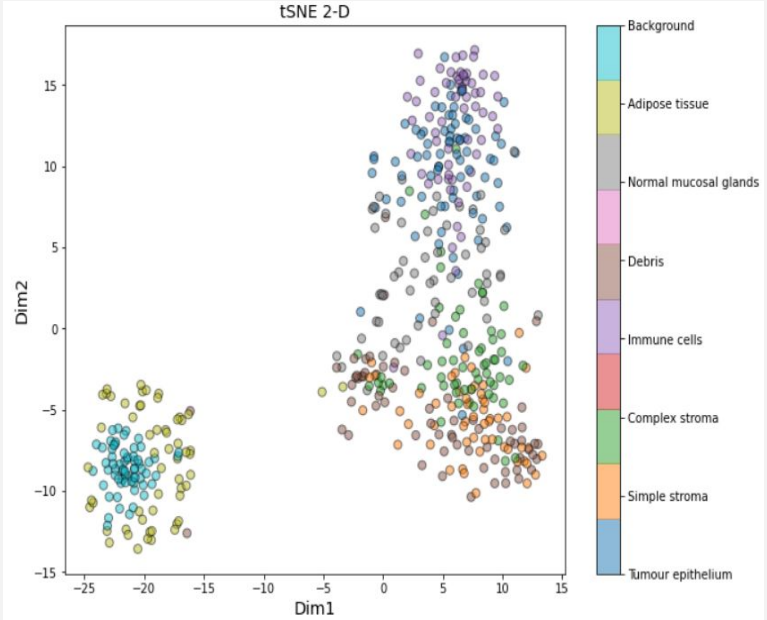
x_i → Reshape (Flatten) → $Z_i \in \mathbb{R}^{67500}$ → PCA → $Z_i \in \mathbb{R}^{256}$ → Neural Network Classifier → argmax → $y_i \in \{1,2, \dots, 8\}$

(150, 150, 3)

(67500,)

(256,)





- Dimensionality reduction: reshape to 67,500-D, then PCA, resulting in a 256-D feature space.

- Classifier: Neural Network dimension reduction to 128-D by ‘relu’ (fully connected), and Classification by ‘softmax’ (trained for epochs=30, verbose=1, fully connected).

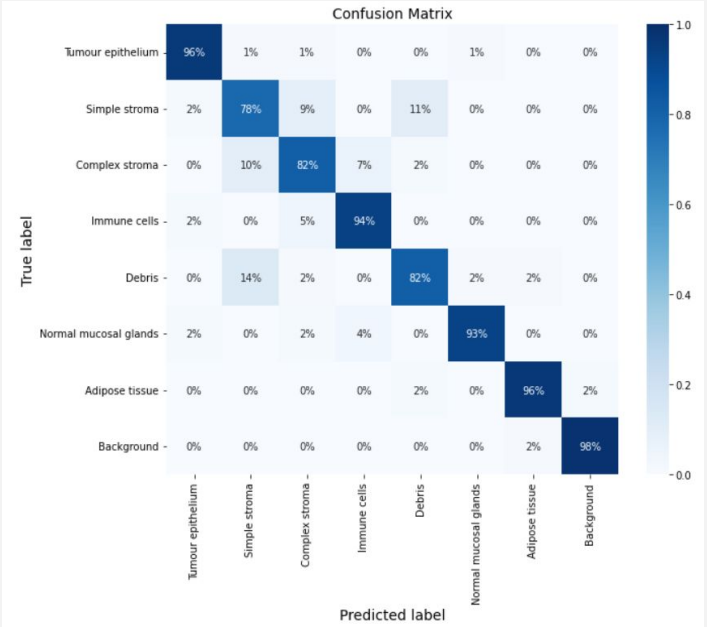
- tSNE: reduction from 256-D to 2-D.

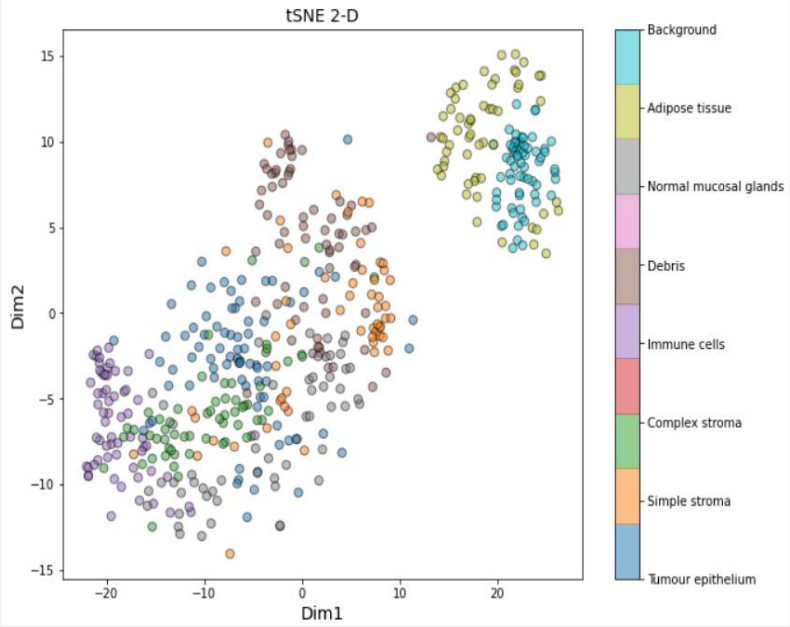
Experiment 7

Training accuracy: 99.9%
Validation accuracy: 90.2%

x_i → VGG →  → Reshape (Flatten) → $Z_i \in \mathbb{R}^{8192}$ → SVM classifier → $y_i \in \{1,2,3, \dots, 8\}$

(150, 150, 3) **(4, 4, 512)** **(8192,)**





- Dimensionality reduction: VGG 16 (trained for epochs=1, batch_size=64, verbose=1) to 4x4x512 and then flattening, resulting in a 8192-D feature space.
- Classifier: SVM classifier (trained for kernel = 'linear', C=1)
- tSNE: reduction from 8192-D to 2-D.

Experiment 8

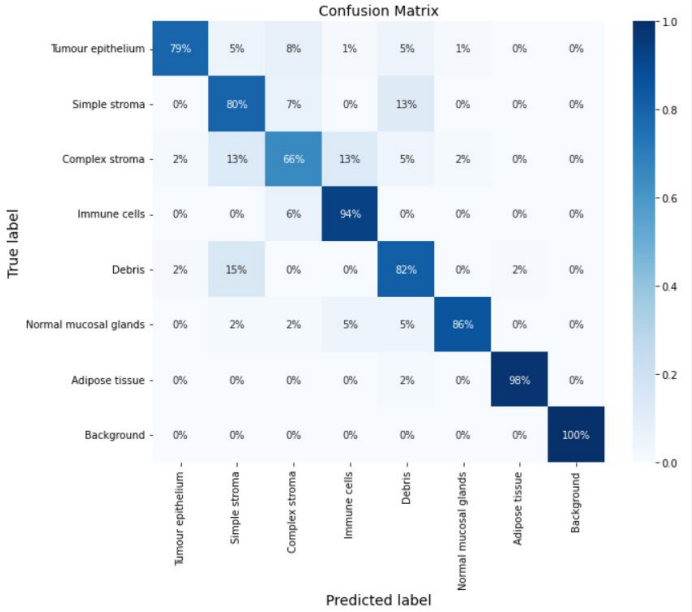
Training accuracy: 93.8%
Validation accuracy: 85.4%

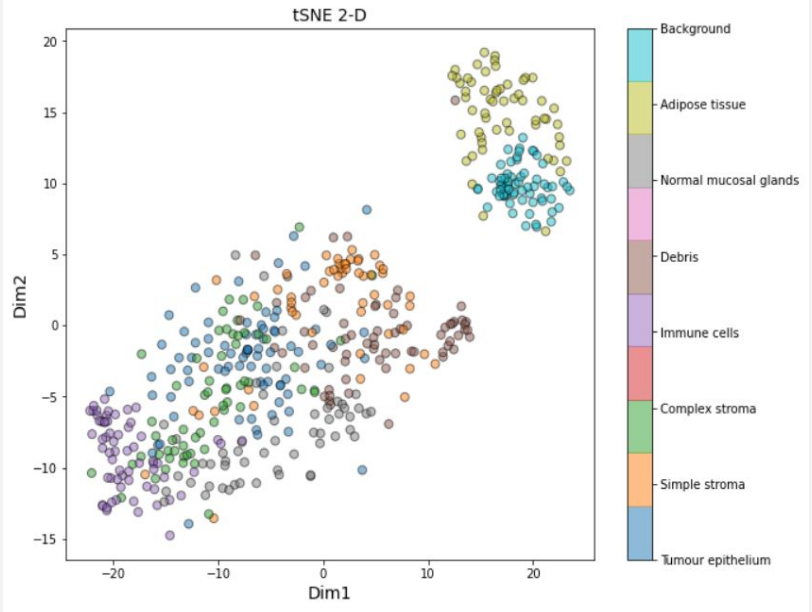
$\mathcal{X}i$ → VGG →  → Reshape (Flatten) → $Zi \in \mathbb{R}^{8192}$ → Softmax Classifier → argmax → $yi \in \{1,2,3, \dots, 8\}$

(150, 150, 3)

(4, 4, 512)

(8192,)





- Dimensionality reduction: VGG 16 (trained for batch_size=64, epochs=25, verbose=1) to 4x4x512 and then flattening, resulting in a 8192-D feature space.

- Classifier: Softmax Classifier (trained for epochs=25 , verbose=1, fully connected).

- tSNE: reduction from 8192-D to 2-D.

Experiment 9

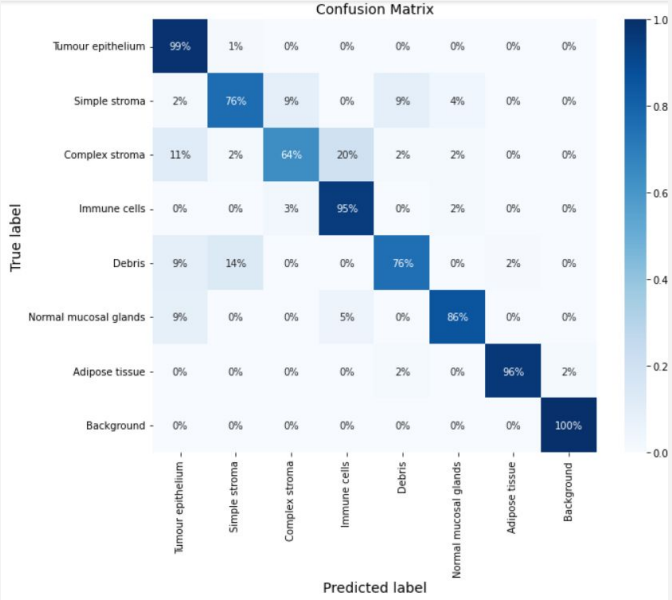
Training accuracy: 93.6%
Validation accuracy: 87%

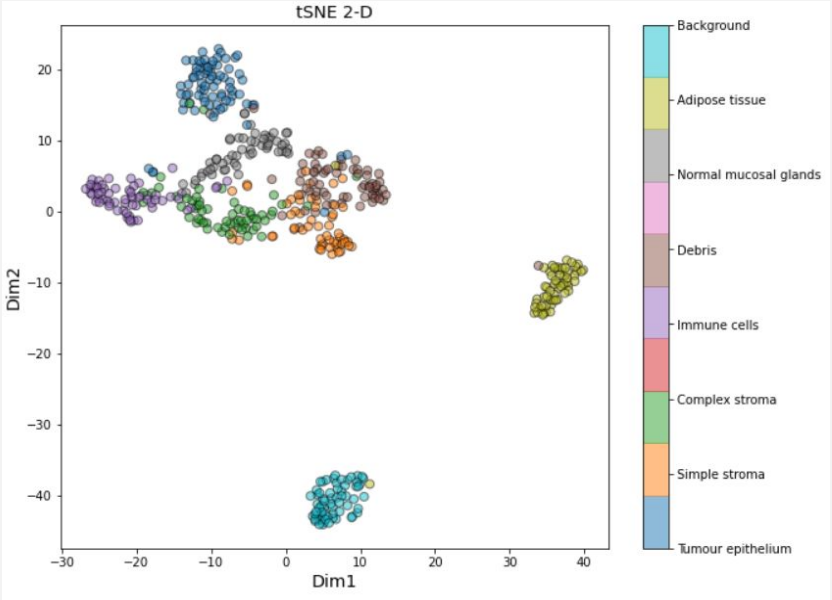
xi → VGG →  → Reshape (Flatten) → $Zi \in \mathbb{R}^{8192}$ → Neural Network Classifier → argmax → $yi \in \{1,2,3, \dots, 8\}$

(150, 150, 3)

(4, 4, 512)

(8192,)





- Dimensionality reduction: VGG 16 (trained for batch_size=64, epochs=30, verbose=1) to 4x4x512 and then flattening, resulting in a 8192-D feature space.

- Classifier: Neural Network dimension reduction to 128-D by ‘relu’ (fully connected), and Classification by ‘softmax’ (trained for 30 epochs, batch_size=64, verbose=1, fully connected).

- tSNE: reduction from 128-D to 2-D.

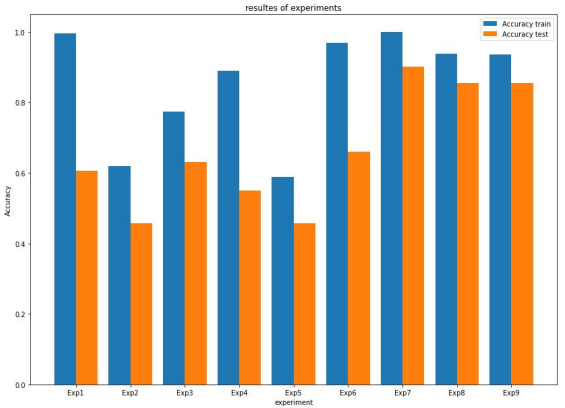
Results Table of the all models:

The model that has shown the best results is the 9th experiment. We can assume that because the VGG 16 model is best for dimensionality reduction for images data, while keeping most of the variance of the data. The VGG 16 model **use lower number of hyper parameters** and composed of 16 layers as a deep neural network with an effective mechanism for dimensionality reduction without changing the quality of the data.

In addition, the neural network **is the best classification** to recognize underlying relationships in a set of data. In our project the neural network used 2 activation functions; 'relu' and 'softmax', in order to reduce dimensions as well, and therefore, **helped with the classification**.

The 8th experiment has shown similar results because the softmax classification is often used as the last activation function of a neural network to normalize the output of a network.

Dimensionality reduction	Classification	train accuracy (%)	test accuracy (%)
Downsample image	SVM	0.996	0.606
Downsample image	Softmax classifier	0.708	0.55
Downsample image	Neural Network	0.616	0.508
PCA	SVM	0.888	0.55
PCA	Softmax classifier	0.589	0.458
PCA	Neural Network	0.969	0.66
VGG	SVM	0.999	0.902
VGG	Softmax classifier	0.938	0.854
VGG	Neural Network	0.936	0.87



Further Recommendations:

1. We recommend to use the **VGG 16 model for classification with images data**, due to the good results we accomplished in the 9th experiment.
2. In order to receive even better results, we will recommend **to use more layers** in the neural network that reduce more dimensions resolving more accurate results.
3. We recommend to train the model on a **bigger size sample** and on a different images to prevent overfitting of the model.
4. We believe that **the data should be divided into 3 groups**: train, test and dev, instead of train and test only, in order to prevent overfitting again.