The Cancer Genome Atlas
Kidney Clear Cell Carcinoma

Alexander Faché
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA, USA
afache3@gatech.edu

Samuel Lovejoy
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA, USA
slovejoy6@gatech.edu

***Index Terms*—The Cancer Genome Atlas (TCGA), kidney cancer, necrosis, stroma, tumor, histopathological, medical image processing, Reinhard's Method, data augmentation, hand crafted feature extraction, supervised learning**

I. Introduction

This document will perform an in-depth analysis of key components related to clinical decision support for the The Cancer Genome Atlas project presented in ECE4783–Introduction to Medical Image Processing. First, a discussion showing the importance of TCGA and the use of histopathological images from cancer patients will set the stage. Following this discussion, a literature review to learn more about feature extraction and implementation methods will be discussed. Next, explanations of the preprocessing steps required to ready our dataset for testing and training will be stepped through. To conclude this document, feature selection, implementation, and initial results will be presented.

II. The Cancer Genome Atlas

A. Overview

In order to support clinical decisions among cancer patients, a model for diagnosis and prognosis of cancer samples will be developed. We observe the image processing pipeline: normalization, feature extraction and selection, and ultimately prediction modeling. In this document, we will discuss applying image processing techniques to cancer histopathological images in order to develop objective, reproducible decision support for the diagnosis and prognosis of kidney cancer.

B. Data

The provided data set for undergraduate students contains quality controlled tissue slide image samples of three stages of cancer: necrosis, stroma, and tumor for kidney clear cell carcinoma cells. These stages of cancer will be our labels and desired output for our trained decision support models. Necrosis classification is a result of premature death of cells. Stroma classification is healthy connective tissue cells. Tumor classification results from abnormal cell divisions that uncontrollably destroy body tissue. Fig. 1 shows sampled images of all three classes.

ECE 4783 Introduction to Medical Image Processing, Spring 2020.

The quality control stage of the preprocessing began with 1000 original whole slide images (20,000 x 40,000 pixels). These images were subject to 512 x 512 pixel sub-section cropping to remove tissue folding, invalid stains, and empty regions of interest (ROI). In the end, 100 images of each class were presented.

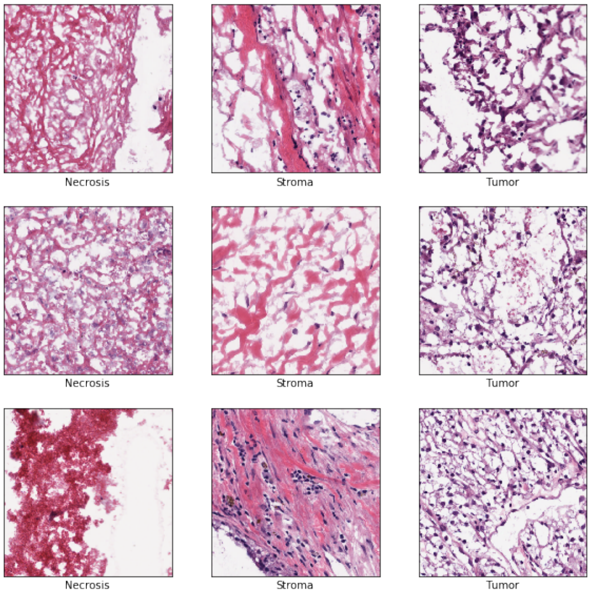

Fig. 1. Sampled images that have undergone ”quality control”.

C. Goal

Given the provided data set that has undergone quality control, the goal of the image processing pipeline is to create a model that actively supports clinical decision making with high precision. The ability the accurately distinct between necrosis, stroma, and tumor classifications will drive this precision.

III. Literature Review

A. Overview

In order to gain more insight into the process of feature extraction, several papers were analyzed. The following discussion looks at color, texture, and morphological based features.

B. Color Based Features

1) Automatic Batch-Invariant Color Segmentation of Histological Cancer Images [1]: Due to variations in tissue specimen preparation, staining, and imaging, color distribution among different samples may vary. A method for resolving such differences is the batch effect where users of the equipment can incorporate domain knowledge during the segmentation process. However, human error was determined to lower objectivity, reproducibility, and speed. With a novel color normalization scheme and domain expertise, a system resistant to batch effects was developed. The proposed solution normalizes images over the color map instead of over the pixels. The issue with pixel level normalization is that it tends to lead to distorted colors in the resulting normalized image. Normalizing over the color map extracts the unique colors however, since it does not look at individual pixels, it does not include the frequency of colors. Results show that color map normalization was more accurate than the all pixels method.

2) Histological image feature mining reveals emergent diagnostic properties for renal cancer [2]: Using colorspaces from 28 different Gabor filters, this paper examines renal cancer images, traditionally a very difficult type of cancer to work with. Each filter and filtered image has a unique energy and entropy, which proved to be one of the highest weighted features. Unlike many other papers in the medical imaging field, this paper opts to classify beyond malignant and benign, but performs prediction modeling with non-binary outcomes. Fractal methods are used for texture features, while Voronoi diagrams are used for topological features.

C. Texture Based Features

1) Automatic Classification of Pathological Prostate Images Based on Fractal Analysis [3]: Standard texture based features are rated using the Gleason grading system. Using a novel texture analysis techniques based on a fractal, detection rates for pathological prostate images were improved compared to previous multiwavelets, Gabor filters, and GLCM methods. The fractal dimension feature set is smaller, however, still effective according to the Gleason grading system.

2) Automated classification of brain tumor type in whole-slide digital pathology images using local representative tiles [4]: Coarseness versus fineness, unlike some other texture features, is very difficult to distinguish by histopathologists. Using a dataset of brain tumor images to classify cancer subtypes, this paper segments its images into tiles, then performs typical analysis on the tiles and weights the tile proper and its neighboring regions. The results of this paper boasted very high accuracy ratings in both of its examined cases of cancer subtypes with an accuracy rating of over 95 percent.

D. Morphological Based Features

1) Mitosis detection in breast cancer pathology images by combining handcrafted and convolutional neural network features [5]: Hand crafting features is a time consuming process and it is sometimes difficult to get an adequate amount. With increasing amount of data available, more of these processes can be automated. In order to assist hand crafted feature creation, convolutional neural networks (CNN) aim to learn features that could otherwise not be represented through hand crafted means. It is important to note that hand crafted features are not being ignored as they better capture domain specific applications. The development of a lightweight CNN cascaded with hand crafted features is a faster approach and less computationally expensive than a stand alone CNN.

2) Object-and Spatial-Level Quantitative Analysis of Multispectral Histopathology Images for Detection and Characterization of Cancer [6]: This thesis follows a dataset of breat cancer images by creating multiple classes for object recognition. Multispectral analysis on nuclear features did not seem to provide as significant features as the non-nuclear features. Feature extraction for this thesis is very well documented, which provided the basis for many of the texture and morphological features we use in this project. While this thesis did not have as many novelties as the other papers published, its documentation is much more thorough.

IV. Module 1 - Image Preprocessing

A. Overview

In the order to feed in our training samples to a model we first need to preprocess our image data. Preprocessing is important as it serves to eliminate noise, perform any clean ups, as well as create more samples. The following sections will outline Normalization IV-B and data augmentation IV-C steps that were applied as well as the results IV-D.

B. Normalization

First color normalization will be applied to remove any side effects of lighting. The proposed implementation is Reinhard's Method which uses the CIELAB color space [7] [8]. The CIELAB color space is represented by:

- L^* - lightness from black to white
- a^* - lightness from green to red
- b^* - lightness from blue to yellow

The implementation makes use of a target image which is used as a reference image. This image contains the distribution of color (RGB channels) that is desired to be mimicked by the source image. Because $l\alpha\beta$ is a transform of the LMS color space, RGB to LMS then LMS to $l\alpha\beta$ transformations must be performed. First, the source image is transformed from the RGB color space to LMS color space via (1):

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1967 & 0.7244 & 0.07802 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{1}$$

Then the LMS image is transformed to $l\alpha\beta$ color space via (2):

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \tag{2}$$

Once in LAB color space, the mean is subtracted from each channel via (3):

$$\begin{aligned} l^* &= l - \bar{l} \\ \alpha^* &= \alpha - \bar{\alpha} \\ \beta^* &= \beta - \bar{\beta} \end{aligned} \tag{3}$$

Then the data is scaled by the respective standard deviations (4):

$$\begin{aligned} l^{'} &= \frac{\sigma_t^l}{\sigma_s^l} l^* \\ \alpha^{'} &= \frac{\sigma_t^\alpha}{\sigma_s^\alpha} \alpha^*, \\ \beta^{'} &= \frac{\sigma_t^\beta}{\sigma_s^\beta} \beta^* \end{aligned} \tag{4}$$

where σ_t stands for the target and σ_s stands for the source standard deviations. Finally, the modified source image is transformed from $l\alpha\beta$ to LMS via (5):

$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & -2 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{3} & 0 & 0 \\ 0 & \frac{\sqrt{6}}{6} & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} \tag{5}$$

Then returned to RGB color space via (6):

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 4.4679 & -3.5873 & 0.1193 \\ -1.2186 & 2.3809 & -0.1624 \\ 0.0497 & -0.2439 & 1.2045 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix} \tag{6}$$

Before applying Reinhard's Normalization on tissue cells where it may be difficult to determine the immediate effect, the normalization process was applied on a test image.

It is clear from Fig. 2 that the darkened sky and grass from the target image (top) was successfully applied to the source image (middle) resulting in the normalized source image (bottom).

After validating Reinhard's Normalization on the test image, the normalization technique was applied using target images Fig. 3, Fig. 5, Fig. 7. The resulting sample normalizations are displayed in Fig. 4, Fig. 6, Fig. 8. From top to bottom, necrosis, stroma, and tumor samples are displayed.

Using the three target images applied to the entire 300 image data set results in 897 normalized images. The target image is removed from the normalization process.

C. Data Augmentation

In order to increase the number of samples without collecting more tissue, data augmentation is applied. In machine learning, the name of the game is data. Copious amounts of data are not easy to come by therefore cropping, flipping, and rotating the normalized images will help to create more samples.

Cropping: The normalized images are 512 x 512 pixels. 224 x 224 pixels sub regions will be taken as this is a common input layer size to convolutional neural networks (CNNs) if the approach is to be taken in the future. Five sub regions are selected at random from each sample image.

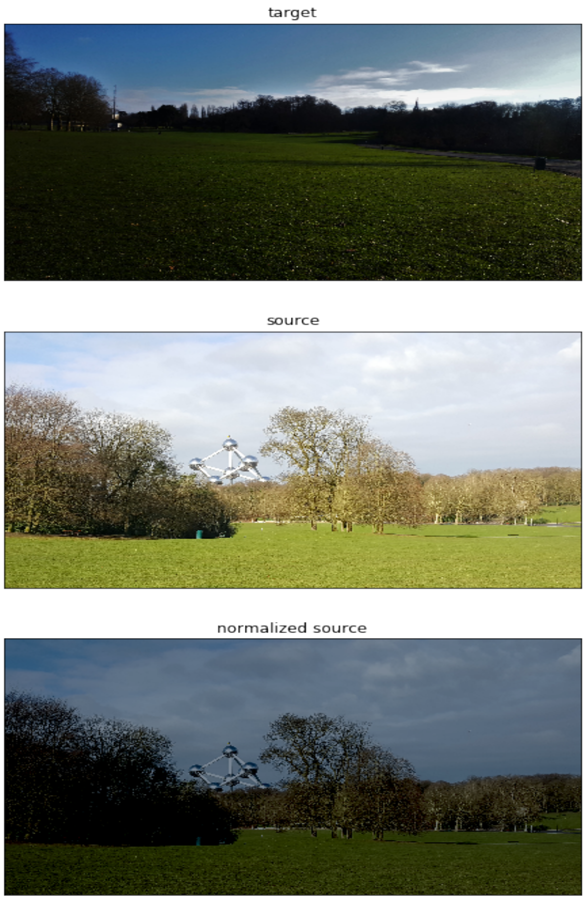

Fig. 2. Example output of Reinhard's Normalization.

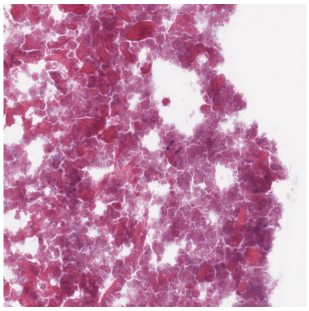

Fig. 3. Necrosis as target.

Flipping: To slightly alter the cropped images, independent vertical and horizontal flips will be performed. A vertical flip of an image is created by flipping the rows of an image shown via (7). Given an image I:

$$I_{new}[:: -1, :, :] = I[:, :, :] \tag{7}$$

A horizontal flip of an image is created by flipping the columns of an image via (8):

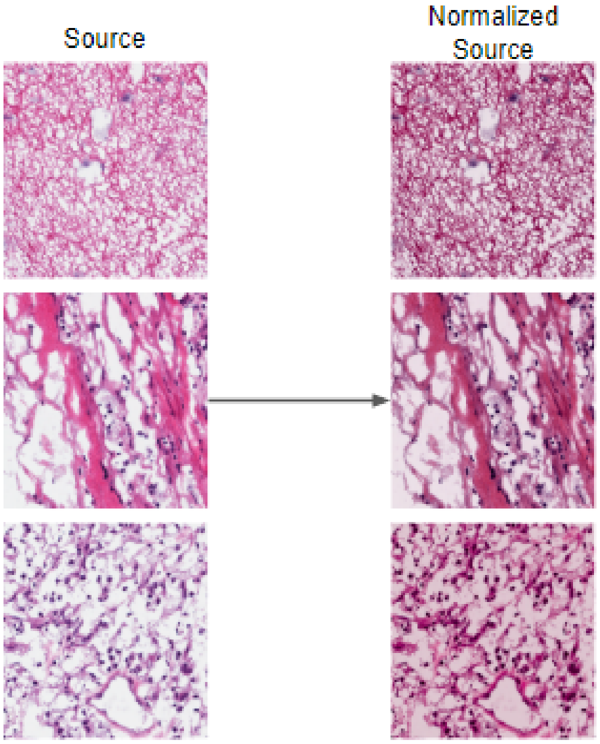

Fig. 4. Output from necrosis normalization.

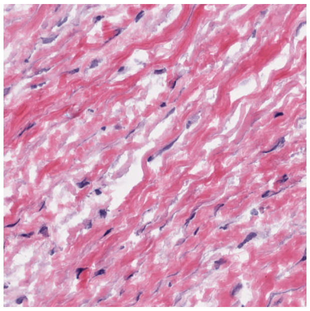

Fig. 5. Stroma as target.

$$I_{new}[:, :: -1, :] = I[:, :, :] \tag{8}$$

Rotating: Another alteration that was employed is rotation. A 90^o counter clockwise rotation is the product of first transposing the image then flipping the rows via (9)-(10):

$$I_{temp} = I^T \tag{9}$$

$$I_{new}[:: -1, :, :] = I_{temp} \tag{10}$$

A 180^o counter clockwise rotation is the product of first flipping the rows then flipping the columns via (11)-(12):

$$I_{temp}[:: -1, :, :] = I \tag{11}$$

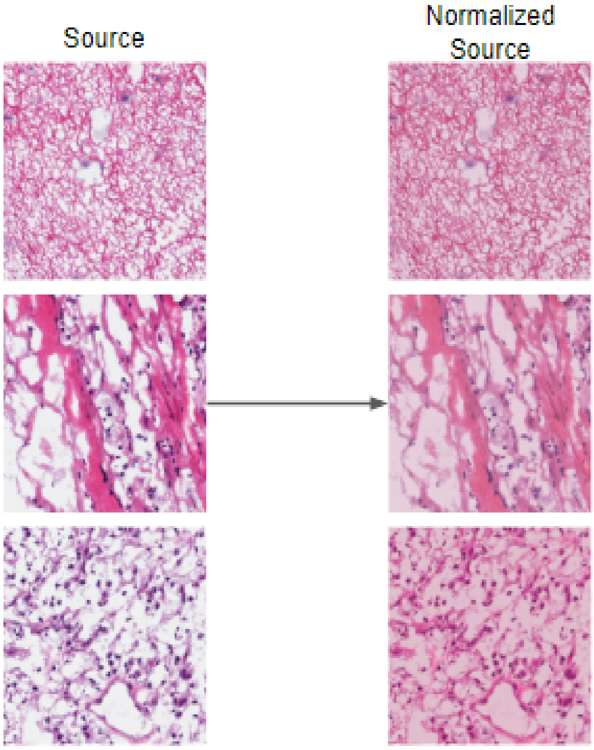

Fig. 6. Output from stroma normalization.

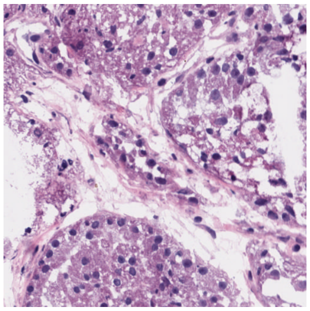

Fig. 7. Tumor as target.

$$I_{new}[::, :: -1, :] = I_{temp} \tag{12}$$

A 270^o counter clockwise rotation is the product of first transposing the image then flipping the columns via (13)-(14):

$$I_{temp} = I^T \tag{13}$$

$$I_{new}[:, :: -1, :] = I_{temp} \tag{14}$$

D. Results

Through two flips and three rotations, each image now has six new samples based on the cropped sub region (five augmented + one original). This process of selecting a random sub region, applying flipping, then rotating is applied a total of five times to result in 30 (5 * 6) samples for each original

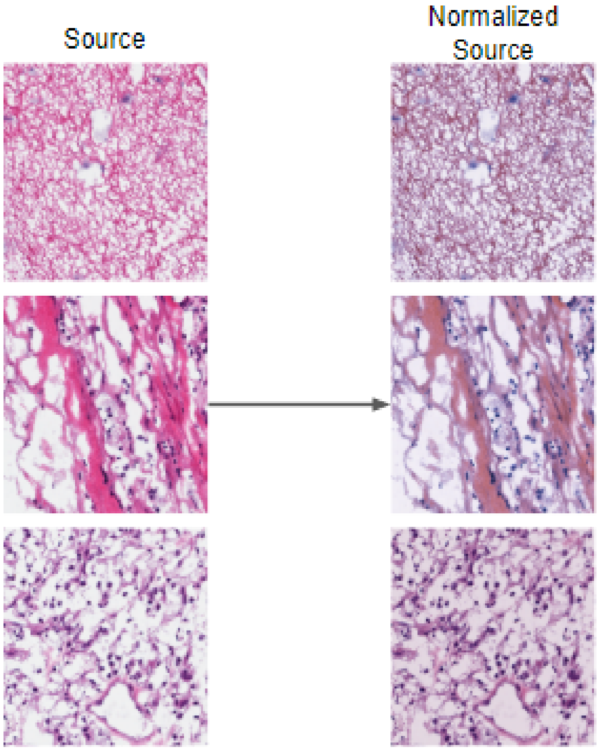

Fig. 8. Output from tumor normalization.

normalized image. Applying to all 897 images results in 26,910 images of dimension 224 x 224 pixels. A sample output for one iteration is shown in Fig. 9-Fig. 12.

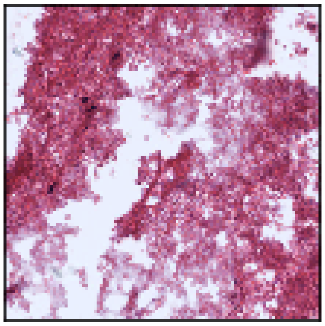

Fig. 9. Original normalized image selected for augmentation.

V. Module 2 - Feature Extraction and Selection

A. Overview

In Section IV, preprocessing and data augmentation were discussed in order to clean up the data set and create more samples. In this section, these raw images will be converted to usable features to train and test machine learning models.

B. Description of Color Based Features

1) Color Spaces:

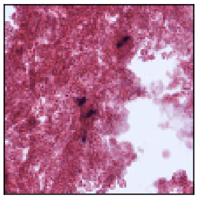

Fig. 10. Cropped sub region.

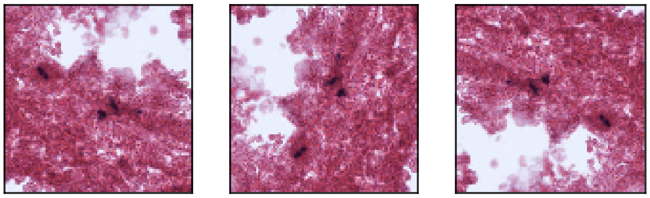

Fig. 11. Result from 90^o (left), 180^o (center), 270^o (right) counter clockwise rotations.

- RGB
- HSV
- LAB

2) Color Channel Properties: Each color space consists of three different color channels represented by a matrix I with r rows and c columns. From each individual color channel the following properties were extracted using $scipy.stats.describe$.

- Minimum value

$$\min_{r,c} I(r,c) \tag{15}$$

- Maximum value

$$\max_{r,c} I(r,c) \tag{16}$$

- Mean value

$$\mu = \frac{1}{N}\sum_{r,c} I(r,c) \tag{17}$$

- Variance

$$\sigma^2 = \frac{1}{N}\sum_{r,c} (I(r,c)-\mu)^2 \tag{18}$$

- Skewness
  - measure of a lack of symmetry
  - degree of symmetry

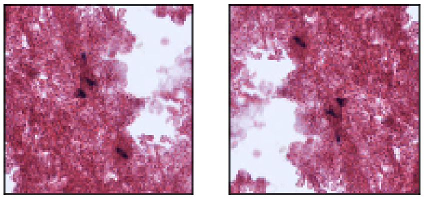

Fig. 12. Result from vertical (left) and horizontal (right) flips.

- $\widetilde{I}$ is the median

$$\frac{3*(\mu-\widetilde{I})}{\sigma} \tag{19}$$

- Kurtosis (Fisher)
  - measure of the pointedness of peak
  - degree of peakedness

$$E\left[\left(\frac{I-\mu}{\sigma}\right)^4\right] \tag{20}$$

- Number of pixels per channel

$$N = r*c \tag{21}$$

C. Analysis of Color Based Features

We elect to perform similar statistical analyses across multiple color dimensions. These include maximum and minimum values, ranges, mean, variance, and skewness for each color space. The purpose of the aforementioned statistical analyses is to identify outliers, as in theory due to the Reinhard normalization process many of these statistics should be very similar image to image. In addition, the red stain applied when capturing the tissue images already does much of the work for us in the color space.

D. Description of Texture Based Features

1) Grey-Level Co-Occurance Matrix properties: GLCM = C(i, j) is determined with $skimage.feature.greycomatrix$.

- Contrast

$$\sum_{i,j} C(i,j)(i-j)^2 \tag{22}$$

- Dissimilarity

$$\sum_{i,j} C(i,j)|i-j| \tag{23}$$

- Homogeneity

$$\sum_{i,j} \frac{C(i,j)}{1+(i-j)^2} \tag{24}$$

- ASM

$$ASM = \sum_{i,j} C^2(i,j) \tag{25}$$

- Energy

$$\sqrt{ASM} \tag{26}$$

- Correlation

$$CORR = \sum_{i,j} C(i,j)\frac{(i-\mu_i)(j-\mu_j)}{\sigma_i\sigma_j} \tag{27}$$

- Inertia

$$\sum_{i,j}(i-j)^2 C(i,j) \tag{28}$$

- Entropy

$$-\sum_{i,j} C(i,j)\log_2 C(i,j) \tag{29}$$

- MaxProb

$$\max_{i,j} C(i,j) \tag{30}$$

- Cluster Shade

$$sgn(A)|A|^{1/3} \tag{31}$$

- Cluster Prominance

$$sgn(B)|B|^{1/4} \tag{32}$$

$$A = \sum_{i,j} \frac{C(i,j)(i+j-2\mu)^3}{\sigma^3\sqrt{2(1+CORR)}^3} \tag{33}$$

$$B = \sum_{i,j} \frac{C(i,j)(i+j-2\mu)^4}{4\sigma^4(1+CORR)^2} \tag{34}$$

These properties are discussed in [6]- [9].

2) Shannon Entropy:

$$-\sum_i p_i \log_2(p_i) \tag{35}$$

3) Signal to Noise Ratio:

$$10\log_{10}\left(\frac{P_{signal}}{P_{noise}}\right) \tag{36}$$

4) Compression Ratio:

$$CR = \frac{filesize(CompressedFile)}{filesize(OriginalFile)} \tag{37}$$

Laplacian of Gaussian

$$\begin{aligned} LoG = conv2D(Laplacian, ksize = 3,\\ conv2D(Gaussian, ksize = 3, image)) \end{aligned} \tag{38}$$

E. Analysis of Texture Based Features

We primarily analyze texture based features by transforming into different domains. Two helpful transforms, not listed above, are the Discrete Cosine Transform (DCT) and Fast Fourier Transform (FFT) both of which tell us many things about the texture of the image. The DCT, often used in image compression, represents the image as a sum of cosines. The more nonzero coefficients of the cosine terms, the more complex the texture of the image is. This is also reflected in the entropy of the image. The FFT puts the image into the frequency domain, which tells us which parts of the image change quickly (i.e. pure black to pure white) and which change slowly (i.e. red to darker red), another useful component for texture analysis. In the future, we will use an autoencoder to further simplify the images.

F. Description of Morphological Based Features

1) Symmetry: Let I_v represent a vertically flipped image and I_h represent a horizontally flipped image.

- Vertical MSE

$$MSE_v = \frac{1}{N}\sum(I-I_v)^2 \tag{39}$$

- Horizontal MSE

$$MSE_h = \frac{1}{N}\sum(I-I_h)^2 \tag{40}$$

2) Area: The total number of pixels in the image object.

$$Area = \sum_{n,m} \Omega(n,m) \tag{41}$$

3) Object Centroids:

$$\bar{x} = \sum_{n,m} n\Omega(n,m) \tag{42}$$

$$\bar{y} = \sum_{n,m} m\Omega(n,m) \tag{43}$$

4) Major and Minor Axis Lengths:

$$major = 2\sqrt{2}\sqrt{m_{xx} + m_{yy} + \sqrt{(m_{xx} - m_{yy})^2 + 4m_{xy}^2}} \tag{44}$$

$$minor = 2\sqrt{2}\sqrt{m_{xx} + m_{yy} - \sqrt{(m_{xx} - m_{yy})^2 + 4m_{xy}^2}} \tag{45}$$

given

$$m_{xx} = \frac{1}{M}\sum_{M}(x_i - \bar{x})^2 + \frac{1}{12} \tag{46}$$

$$m_{yy} = \frac{1}{M}\sum_{M}(y_i - \bar{y})^2 + \frac{1}{12} \tag{47}$$

$$m_{xy} = \frac{1}{M}\sum_{M}(x_i - \bar{x})(y_i - \bar{y}) \tag{48}$$

where M is the number of pixels within the object.

5) Eccentricity: A measure of conic sections that shows how elliptical an object is–how far an object is from being circular.

$$ecc = \frac{2\sqrt{(major/2)^2 - (minor/2)^2}}{major} \tag{49}$$

6) Orientation: A measure of the angle between the major elliptical axis and the image's original x-axis.

$$\theta_O = \arctan\left(\frac{m_{yy} - m_{xx} + \sqrt{(m_{yy} - m_{xx})^2 + 4m_{xy}^2}}{2m_{xy}}\right) \tag{50}$$

7) Convex Area: The area of the convex hull of the object (from MATLAB).

$$ConvexArea = \sum_{n,m} ConvHull\Omega(n,m) \tag{51}$$

8) Convex Deficiency: The pixels within the convex hull that are not within the object (exuding spikes).

$$ConvexDef = \frac{ConvexArea - Area}{Area} \tag{52}$$

9) Solidity: The fraction of pixels within the convex hull computed by MATLAB that are also within the object.

$$Solidity = \frac{Area}{ConvexArea} \tag{53}$$

10) Perimeter: The distance around the boundary of the object. The last coordinate is identical to the first.

$$Per = \sum_{n=1}^{N} \sqrt{(x(n+1) - x(n))^2 + (y(n+1) - y(n))^2} \tag{54}$$

11) Equivalent Diameter: The diameter of the circle that could enclose the object.

$$EquivDiameter = 2\sqrt{\frac{\pi}{Area}} \tag{55}$$

12) Sphericity: The ratio of the smallest circle that encloses the object over the largest circle that can be enclosed by the object.

$$Sphericity = \frac{min(EquivDiameter/2)}{max(EquivDiameter/2)} \tag{56}$$

13) Proportion White Pixels:

$$\frac{\sum I > 220}{N} \tag{57}$$

14) Proportion Black Pixels:

$$\frac{\sum I < 100}{N} \tag{58}$$

G. Analysis of Morphological Based Features

We chose to invest more features into morphological features than texture and color space features. The most important of these are edge detection filters such as the Sobel, Canny, and Prewitt filters. Many of the aforementioned features are derived from MATLAB's built-in object analysis using the *regionprops* function. One of our primary goals for morphological feature extraction was to identify the locations and number of nuclei within the hisopathology images. At a first glance, it seems that performing texture analysis of the results of edge detection is a strong way of locating nuclei. Taking the FFT of the Perimeter as done by L. Bouchard in her PhD thesis outperformed the Hough Circle Transform [6]. This is believed to be since the nuclei are so small that the Hough Circle Transform becomes inaccurate.

H. Feature Scaling

Feature scaling is important in assisting machine learning models. Typically models perform better and converge faster when features are of similar magnitude and are normally distributed [10]. *sklearn.preprocessing* offers several feature scalers one of which is the MinMaxScaler. For a feature F, MinMaxScaler subtract the minimum value of the feature then divides by the feature's range (59):

$$\begin{aligned} m_n &= \min F \\ m_x &= \max F \\ F_{scale} &= \frac{F - m_n}{m_x - m_n} \end{aligned} \tag{59}$$

MinMaxScaler preserves the original distribution and returns scaled features in the range $[0, 1]$.

I. Dimensionality Reduction

1) Principle Component Analysis (PCA): The act of dimensionality reduction is to reduce the number of features used to describe the data set. Doing so can reduce complexity, avoid irrelevant features, and reduce computation needs. PCA uses singular value decomposition in order to transform the original features to Z space. These new features, called principal components, are optimized in order to maximize the variance. By doing so they also become linearly independent allowing us to ignore redundant features. In order to determine th optimal number of principal components to keep, a scree plc is used Fig. 13. A scree plot shows the individual recovere variance for each principal component in descending orde Along with a plot of the cumulative recovered variance Fig. 14, the number of principal components can be selected t meet a desired recovered variance percentage. Note that 100 recovered variance is equivalent to using the original numbe of features and therefore inappropriate for dimensionalit reduction. Equation (60) shows the transformation from dat set X to Z space using the top K principle components:

$$U, S, V = SVD(X)$$
$$Z = X \bullet V_{0...K} \quad (60$$

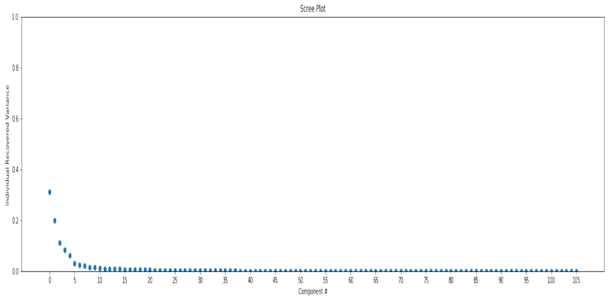

Fig. 13. Scree plot.

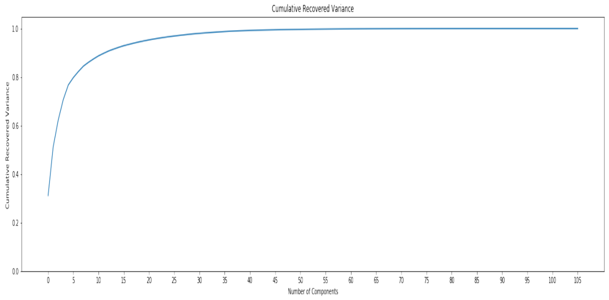

Fig. 14. Cumulative recovered variance.

2) Correlation Matrix: After performing PCA we expec to have completely independent features in order to avoid redundant features/information. For example, let's say we have two features: height in inches and height in centimeters. These two features are proportional to each other by a scaling factor of 2.54 centimeters per inch and are therefore linearly dependent. For simplicity sake we can remove one of these features without losing any information. This is the hindsight for using PCA. To show that all of our features are independent, a correlation matrix can be constructed for the original features and the Z space features. An identity matrix represents completely linearly independent features. Before and after are correlation matrices are shown in Fig. 15-Fig. 16. Using only 39 principal components, 99% of the original 106 components' variance is recovered.

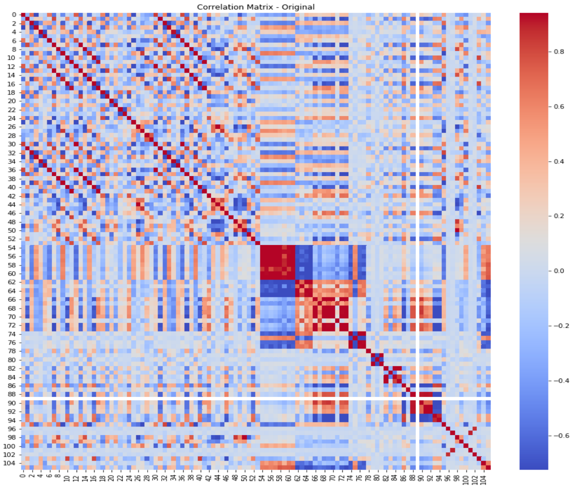

Fig. 15. Correlation matrix of original features.

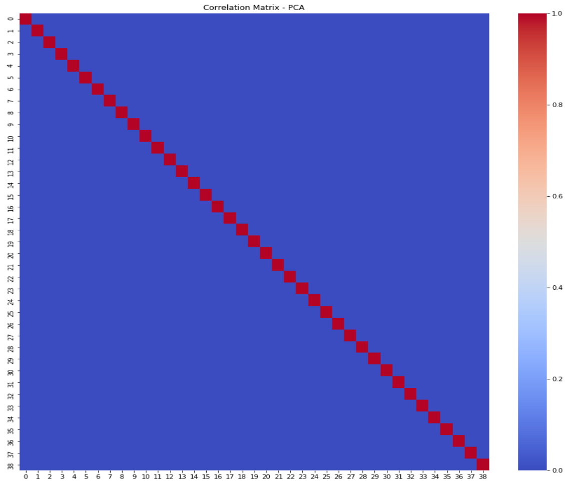

Fig. 16. Correlation matrix of Z space features.

3) Random Forest: Using a forward selection random forest, the features can be ranked according to importance. Shown in Fig. 17 are the rankings of a subset of features using only 1000 training images. Features 104, 37, 4, 5, 0, 13, 34, 74, 63, and 77 are the top features in terms of importance. These correspond to *proportion of white pixels, L channel minimum value, red channel skewness, red channel kurtosis, red channel minimum value, blue channel maximum value, intensity channel variance, maximum energy, mean homogeneity, and minimum correlation*.

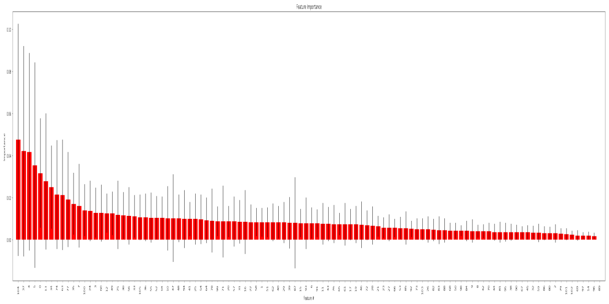

Fig. 17. Random forest feature importance.

VI. Module 3 - Classification for Clinical Decision Making

A. Overview

The final stage of the image processing pipeline consists of classification for clinical decision making. With knowledge of the given histopathological images, implementation of required preprocessing steps, and feature extraction analysis, machine learning models can be trained to differentiate between the three types of tissue stages: necrosis, stroma, and tumor. Several supervised machine learning models for decision boundary creation are discussed below.

1) Multinomial Logistic Regression - One-vs-All: With three classes (necrosis, stroma, and tumor), multinomial logistical regression aims to create a decision boundary (61) for each class to predict the probability of a sample with features θ belonging to class i. When a new input x is fed into the model, a prediction will be made according to (62).

$$h_\theta^i = P(y = i|x; \theta), i \in 1, 2, 3 \tag{61}$$

$$\max_i h_\theta^i(x) \tag{62}$$

B. Expected Results

Current physicians often have tumor accuracy classification rate of 60%. In order to assist human clinical decision making we will aim to top this classification rate for necrosis, stroma, and tumor cancer stages.

References

[1] S. Kothari, J. H. Phan, R. A. Moffitt, T. H. Stokes, S. E. Hassberger, Q. Chaudry, et al., "Automatic batch-invariant color segmentation of histological cancer images," Conf Proc IEEE Int Symp Biomed Imaging, ISBI, pp. 657-660, 2011.

[2] S. Kothari, J. H. Phan, A. N. Young, and M. D. Wang, "Histological image feature mining reveals emergent diagnostic properties for renal cancer," Conf Proc IEEE Bioinform Biomed, BIBM, 2011.

[3] P. Huang and C. Lee, "Automatic Classification for Pathological Prostate Images Based on Fractal Analysis," in IEEE Transactions on Medical Imaging, vol. 28, no. 7, pp. 1037-1050, July 2009.

[4] J. Barker, A. Hoogi, A. Depeursinge, D. L. Rubin, "Automated classification of brain tumor type in whole-slide digital pathology images using local representative tiles," Medical Image Analysis, vol. 30, pp. 60-71, 2016.

[5] H. Wang, A.C. Roa et al, "Mitosis detection in breast cancer pathology images by combining handcrafted and convolutional neural network features". Journal of Medical Imaging, 1(3), 034003 (2014).

[6] L. Boucheron, "Object-and spatial-level quantitative analysis of multispectral histopathology images for detection and characterization of cancer," PhD thesis, University of California, Santa Barbara, 2008.

[7] D. Magee, D. Treanor, D. Crellin, M. Shires, K. Smith, K. Mohee, et al., "Colour Normalisation in Digital Histopathology Images," Proc. Optical Tissue Image analysis in Microscopy, Histopathology and Endoscopy (MICCAI Workshop), pp. 100-111, 2009

[8] E. Reinhard, M. Adhikhmin, B. Gooch and P. Shirley, "Color transfer between images", IEEE Computer Graphics and Applications, vol. 21, no. 4, pp. 34-41, 2001. Available: 10.1109/38.946629.

[9] "GLCM Texture Feature." GLCM Texture Feature, 1 Aug. 2019.

[10] J. Hale, "Scale, Standardize, or Normalize with Scikit-Learn", Medium, 2019. [Online].