Developing a Classifier Model for Lung Tumors in CT-scan Images

Satrajit Basu, Lawrence O. Hall, Dmitry B. Goldgof
Department of Computer Science and Engineering,
University of South Florida,
Tampa, USA
{hall,goldgof}@cse.usf.edu

Yuhua Gu, Virendra Kumar, Jung Choi,
Robert J. Gillies, Robert A. Gatenby
Department of Imaging
H. Lee Moffitt Cancer Center and Research Institute
Tampa, USA

***Abstract*—A CT-scan is a vital tool for the diagnosis of lung cancer via tumor detection. Developing a classifier to make use of the information in CT-scan images could provide a non-invasive alternative to histopathological techniques such as needle biopsy to identify tumor types. Image features extracted from 74 lung tumor objects of CT-scan images are used in classifying tumor types. Classification is done into two major classes of non-small cell lung tumors, Adenocarcinoma and Squamous-cell Carcinoma, each constituting 30% of all lung tumors. In this first of its kind investigation, a large group of 2D and 3D image features which were hypothesized to be useful are evaluated for effectiveness in classifying the tumors. Classifiers including decision trees and support vector machines are used along with feature selection techniques (Wrappers and Relief-F) to build models for tumor classification. Results show that over the large feature space for both 2D and 3D features it is possible to recognize tumor classes with about 68% accuracy, showing new features may be of help. The accuracy achieved using 2D and 3D features is similar with 3D easier to use.**

***Index Terms*—CT, Texture Feature, Support Vector Machine, Leave-One-Volume-Out, Wrapper, Relief-F, Grid Search, 5x2 Cross Validation, F-test**

I. Introduction

A Computed Tomography (CT) scan is a vital medical imaging technique especially in the field of thoracic radiology [2]. A thin slice CT scan of the chest is an essential procedure for lung cancer diagnosis. Previous works [3], [4] have concentrated mostly on the analysis of CT-scan images to detect tumors and other anomalies of the lungs. However, minimal work has been done in attempting to classify tumor classes based on these images for which new ground is broken here. The common practice for determining tumor class is to perform a histopathological analysis on tissue samples obtained by invasive techniques such as a needle biopsy. As time and cost are crucial factors when it comes to the treatment of a lung tumor, an automated image based classifier could act as a precursor to histopathological analysis, thus enabling the kick-starting of class specific treatment procedures.

In conjunction with experts on lung cancer imaging, the largest set of features ever applied to the problem of differentiating lung tumors are explored in this work, in both 2D and 3D. They range from texture features to morphological features to geometric features. The major features are described in the next section. Once the feature values for the tumor object are extracted from the images, they are then used to build datasets on which the different classifier models are built and evaluated.

II. Materials and Methods

A. Data Sets

The data used here are CT-scan images from the Moffitt Cancer Center, Tampa. The images are in the DICOM (Digital Imaging and Communications in Medicine) format. The slice thickness of the acquired CT-images ranged from 3mm to 6mm. CT-scans of 74 patients were used for comparison of classifiers using 2D and 3D features. The class distribution of the lung tumors is 38 Adenocarcinoma and 36 Squamous-cell Carcinoma images. These tumor types, each make-up 30% of all lung tumors [1]. In each of the cases only one tumor was present in the lung. All patient identification information has been removed.

B. Image Pre-processing

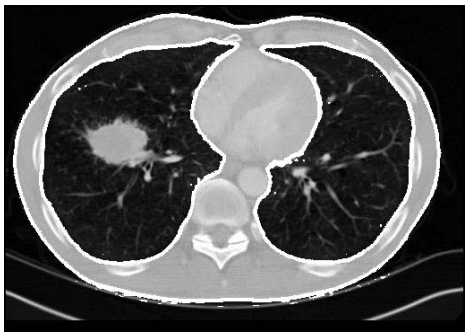
The CT-scan images were processed using the *Lung Tumor Analysis* (LuTA) software suite [5]. The LuTA software does automatic segmentation of the organs in each CT-image slice. Once the segmentation was completed, the tumor was manually indicated by one of the radiologists in the Moffitt Cancer Center or a person with expertise in identifying lung tumors. Once the tumor was identified by marking one point in it, the LuTA software applied a region-growing algorithm to find the tumor boundary across the image sequence. This boundary contained the tumor objects in each slice of the CT-image sequence. Figures 1(a) and 1(b) show the lung with tumor and with the tumor boundary outlined after region growing, respectively.

C. Image Feature Extraction and Feature List

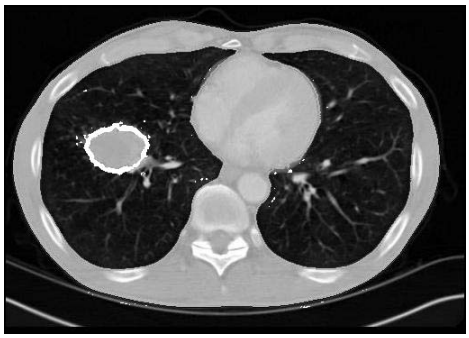
The image feature extraction algorithms were written in C++ and the executables were embedded into the LuTA software. The image feature extraction was done on only the tumor objects. There are 102 2D features and 215 3D features. The 2D/3D features include a combination of texture features, morphological features, geometric features and intensity based features. The performances of classifiers built with 2D and 3D image features were evaluated separately.

The details of the features are provided in the next subsection.

978-1-4577-0653-0/11/$26.00 ©2011 IEEE



(a) Segmentation of CT image to define the lung region



(b) Defining tumor boundary through region growing

D. Feature Measurements

We extracted several types of features for lung cancer classification, such as morphological features, texture features, geometric features and intensity based features. For the purposes of tumor identification and later classification based on aggressiveness, all or some small subset of these features may be useful. Because we have so many features in our study, we will only introduce a few of them. The rest can be found at `http://www.cse.usf.edu/~hall/ctfeatures.doc`. The morphological features include: margin gradient, fractal dimension and Fourier descriptor. The texture features used here are: co-occurrence matrices, run-length analysis, Laws features, and wavelet decomposition, which were similarly used in [6]. The intensity based features include histogram features: mean, standard deviation, energy, entropy, skewness and kurtosis. The lung tumor was delineated from the CT image using the Definiens software (LUTA), and the feature computations were limited to within the lung tumor.

Morphological features. The morphological features were all 2D features.

The *Margin gradient* measures the attenuation value from the centroid of the lesion to the background. It fits data to extract the gradient at the edge of the tumor (in HU/mm). We then measure every 1 degree from spokes emanating from the centroid. From these data we reduce dimensionality further to the mean and standard deviation. The *Fractal dimension* is a useful measure of the morphology of a complex pattern [7]. Fractal geometry is a way to quantify natural objects with a complex irregular structure that are difficult to quantify by regular Euclidean geometrical methods. The study of fractals has been extended to biological structures in past work [8], [9], [10]. The Fractal dimension was applied to quantitatively characterize the complexity of the 2D boundary of the tumor in our case. The *Fourier descriptor* (FD) is another widely used shape descriptor [11]. The FD is obtained by applying the Fourier Transform on a shape contour, where each contour pixel is represented by a complex number.

Texture features. The texture features are available in 2D and 3D.

Co-occurrence matrices. The Co-occurrence matrix [12] is a matrix that contains the frequency of one gray level intensity appearing in a specified spatial linear relationship with another gray level intensity within a certain range. Computation of features requires first constructing the co occurrence matrix, then different measurements [13] can be calculated based on the matrix. The measurements include: contrast, energy, homogeneity, entropy, mean and max probability, etc.

Run-length analysis. Run-length texture features [14] examine runs of similar gray values in an image. Runs may be labeled according to their length, gray value, and direction (either horizontal or vertical). Long runs of the same gray value correspond to coarser textures, whereas shorter runs correspond to finer textures. In our study, texture information was quantified by computing 11 features [15] derived from the run-length distribution matrix. They are

1: Short Run Emphasis (SRE). 2: Long Run Emphasis (LRE). 3: Gray-Level Non-uniformity (GLN). 4: Run Length Non-uniformity (RLN). 5: Run Percentage (RP). 6: Low Gray-Level Run Emphasis (LGRE). 7: High Gray-Level Run Emphasis (HGRE). 8: Short Run Low Gray-Level Emphasis (SRLGE). 9: Short Run High Gray-Level Emphasis (SRHGE). 10: Long Run Low Gray-Level Emphasis (LRLGE). 11: Long Run High Gray-Level Emphasis (LGHGE).

2D image features are useful for capturing surfaces of 2D image objects, however, in many medical imaging applications, discovering the interior structures of the 3D object is necessary. Volume features are often calculated as a series of 2D images and 2D texture features are usually computed for pixels in slices. However this kind of processing will result in losing information across slices. Like in [18], co-occurrence matrices and run-length analysis features can be obtained in 3D, the features are calculated in 13 different directions, with each direction, processing is done by plane instead of slice. Hence, information between slices is not ignored.

Laws features. Laws features [16] were constructed from a set of five one-dimensional filters, each designed to reflect to a different type of structure in the image. These one-dimensional filters are defined as E5 (edges), S5 (spots), R5 (ripples), W5 (waves), and L5 (low pass, or average gray value). By using these 1-D convolution filters, 2D filters are generated by convolving pairs of these filters, such as L5L5, E5L5, S5L5, W5L5, R5L5, etc. Totally we can generate 25 different 2D filters. 3D laws filters were constructed similarly to 2D. 3D filters are generated by convolving 3 types of 1D filter,

such as L5L5L5, L5L5E5, L5L5S5, L5L5R5, L5L5W5, etc. The total number of 3D filters is 125. For the 3D case, after the convolution with the 3D filters for the image, the energy [17] of the texture feature was computed by the following equation: $E = \frac{1}{R} \sum_{i=N+1}^{I-N} \sum_{j=N+1}^{J-N} \sum_{k=N+1}^{K-N} h^2(i,j,k)$, where R is a normalizing factor, I and J, K are image dimensions, h(i,j,k) is derived from the convolution filters and original image. For the 2D case, the above equation is very similar, but without the 3rd (z direction) dimension.

Wavelet decomposition. A wavelet transform decomposes an image into several components iteratively [19] based on the frequency content and orientation. For each iteration, the wavelet decomposition of a 2D image can be achieved by applying the 1-D wavelet decomposition along the rows and columns of the image separately. After several iterations, we will get multiple diagonal, horizontal and vertical components. A wavelet transform of a 3D signal can be achieved by applying the 1-D wavelet transform along all the three directions (x,y,z).

III. Experiments

The extracted feature values formed the base set from which the tumors were classified. For the 2D slices, a certain amount of filtering was done before the data was evaluated as described in the following. Observing the segmentation of the tumor objects we found multiple very small tumor objects, in terms of pixels. These mainly consisted of either tumor fragments or the objects identified in slices marking the beginning of a sequence containing the tumor. Many feature extraction algorithms fail for such tumor objects. Hence, during feature extraction a threshold value of 30 pixels was set for tumor objects. This threshold was identified based on observation. From the 74 volumes a total of 710 (Adenocarcinoma: 347; Squamous-cell Carcinoma: 363) tumor objects from 672 (Adenocarcinoma: 324; Squamous-cell Carcinoma: 348) slices were identified and features were extracted from them. Still, there were a few tumor objects for which one or many of the feature extraction algorithms failed. It was observed that the size in terms of area (measured as number of pixels in the object) for which the feature extraction process failed, varied from one volume to another. Hence a definitive area threshold eludes us at the moment.

However, it was observed that leaving out those objects for which feature extraction was not possible did not result in a choice of random slices within a tumor but instead resulted in, a contiguous sequence of slices containing tumor objects in each of the volumes. Thus successful feature extraction was done on the dataset as follows: for 74 volumes, 675 tumor objects (Adenocarcinoma: 323; Squamous-cell Carcinoma: 352) present in 592 (Adenocarcinoma: 260; Squamous-cell Carcinoma: 332) slices. More filtering of the data was done to select only one tumor object for each slice of the CT-image sequence. This filtering was done by choosing the largest tumor object for each slice in terms of area in pixels. By keeping only the largest tumor object, when there are multiple objects in a single slice, the number of tumor objects under consideration was reduced down to 592 (Adenocarcinoma: 260; Squamous-cell Carcinoma: 332) for 74 volumes. Given the small number of volumes, the evaluation of the classifier was done by performing a *'leave-one-volume-out'* experiment. We also performed a 10-fold cross validation experiment for two reasons. It can give a reasonable estimate of accuracy on unseen data when there is enough data and it avoids potential pitfalls of leave one out.

There is a complication for 2D features. Each instance in a volume represents one tumor object which will be the largest tumor object in a slice (rarely there will be fragments). The prediction for the entire volume is more important than that for each individual tumor object. The leave-one-volume-out experiment involves training the classifier model on examples from all but one volume and testing on the remaining volume. Since each instance in the test set represents a tumor object for the given volume, class prediction for the entire volume is done by means of voting, where class prediction for the majority of objects is taken to be the tumor class. In case of ties, tie breaking was explored by two methods. First by choosing the class of the tumor object on the $(\frac{n}{2}+1)^{th}$ slice in the sequence, where n is the number of slices having a tumor object (i.e. the middle slice). A second method of tie-breaking involved choosing the class of the largest tumor object for the given volume. This mode of tie-breaking was employed since it is reasonable to believe that feature extraction from the largest tumor object for a given volume would contain the maximum possible information. Accuracy by volume for both the tie-breaking techniques will be shown. For each of 74 folds, the training set consisted of 73 tumor volumes and the trained model was evaluated on the one remaining volume. For 10 Fold Cross Validation on 2D features, using the same distribution of volumes in each fold as for 3D features, individual volumes in a single fold were tested against data from the remaining 9 folds. The average accuracy over the folds is reported.

A. Classifier Models

The classifier models evaluated here are *Decision Trees* [20], *Random Forests* [21], *Nearest Neighbor* [22] and *Support Vector Machines* (SVM) [23]. The experiments were performed using the Weka-3.6 data mining tool [24]. For the Support Vector Machine libSVM [25] was used. For feature selection, Relief-F [26] and Wrappers [27] have been employed. Feature Selection was done on the entire training data. The different classifier models and the parameters used for this work are briefly described below.

The Decision Tree used for the experiment was J48, an implementation of the C4.5 Decision Tree [28]. The parameter settings for the J48 were as follows: the confidence factor is set to 0.25.

Random Forests consist of an ensemble of decision trees. In this method, the training data is bagged a specified number of times and then for each subset of instances a decision tree is built. At every decision tree node, a random set of attributes are chosen and the best among them are used as a test. In this

work, the forest contained 200 decision trees and randomly chose $log_2(n) + 1$ features from n total features at each node. The class predicted comes from a vote of the trees.

The Nearest Neighbor algorithm used was the *IB-k* on Weka implementation. It is a modified version of K nearest neighbors. The nearest neighbor search can be done employing brute force linear search or by using other data structures such as KD-trees. In this work the simple linear search algorithm was used. The distance metric was the Euclidean distance. Since the scale of the attributes determines the distance measure, attributes with larger ranges would dominate. Hence, the weka implementation of the algorithm performs normalization on the attributes before measuring the distance metric. The number of nearest neighbors chosen for this particular set of experiments was 5.

Support Vector Machines are based on statistical learning theory [29] and are known to achieve high accuracy on a diverse range of application domains [30]. The idea behind SVMs is to map the input data to a higher dimensional feature space and construct a hyper plane so as to maximize the margin between classes. The hyper plane construction can be reduced to a quadratic optimization problem which is determined by subsets of training patterns that lie on the margin, termed Support Vectors. The hyper plane in the input space is in the form of a decision surface, the shape of which is determined by the kernel chosen. The *Radial Basis Function* (RBF) Kernel was used here. The expression for the RBF kernel is given by $exp(-gamma|u - v|^2)$. All data was scaled to be in the range $[-1, 1]$. The parameter settings for libSVM were as follows: the cost parameter C was set to 15 while gamma was set to 0.009. Support Vector Machines have previously been used efficiently on CT scan image data of the lungs [31] in a Computer-Assisted Detection (CAD) system for automated pulmonary nodules detection in thoracic CT-scan images.

Each of the classifiers described above were first run without any feature selection being employed. Given the large parameter space for both 2D and 3D features and smaller number of examples, it is necessary to perform feature selection. The first feature selection method used was Relief-F, done with the Ranker search algorithm. Relief-F which is *Recursive Elimination of Features* chooses instances at random and changes the weights of the feature relevance based on the nearest neighbor. The Ranker algorithm assigns a rank to each individual feature. The number of features to be chosen for evaluation can be done by means of a parameter. For this work, 50 features were chosen and then the feature space was further reduced to 25 features.

The second feature selection technique employed was Wrapper Feature Selection. This involves evaluating attribute subsets with an underlying classifier model. In this case the underlying model was the same as the classifier being evaluated. That is, if the classifier in use was a Support Vector Machine with an RBF kernel, the underlying classifier for Wrappers was also a Support Vector Machine with an RBF kernel. Feature Selection using Wrappers was done by using Best-First forward selection search. Wrapper Feature selection was not used for Random Forests classifiers. The reason being, both approaches involve selecting of subset of features for evaluation.

B. Feature Merit using F-test

It is of interest to determine whether there is statistically significant improvement if 3D features are selected over 2D features. This is done by means of performing an F-test on results obtained from a 5x2 Cross Validation. The combination of 5x2 Cross Validation with the F-test has been used to compare supervised learning algorithms [32]. The goal here is to, using the same classifier models, compare classifier accuracy using 3D features with that of 2D features.

For comparison, 5x2 Cross Validation was first run for 3D features. Then making use of the volumes in each fold, 2D feature evaluation was done in a manner similar on the same 10 folds. For this evaluation only a single tie-breaking method, choosing the class of the largest tumor object, was employed for 2D features.

IV. Results

A leave-one-volume-out (LOVO) experiment with 2D features using classifiers with the previously described settings was done with the results shown in Table I for 2D features.

In Table I, the average accuracy over slices in the 74 volumes is shown in the first column. The next two columns show the percentage of volumes that were correctly identified. This is the result of voting among the tumor objects that constitute each tumor volume. The fourth column shows accuracy when the tie-breaking for voting is done by choosing the class of the $(\frac{n}{2}+1)^{th}$ slice in the sequence, where n is the number of slices having a tumor object. The fifth column represents accuracy when the method of tie-breaking was done choosing the class of the largest tumor object for the given volume. In the case of 2D features, the number objects of the two classes of tumor were, Adenocarcinoma: 260 and Squamous-cell Carcinoma: 332.

A. Leave one out

In Table I, when the classifiers were run without any feature selection employed, libSVM had relatively higher accuracy than the rest of the models. We see that the most accurate classifier was libSVM after wrappers feature selection at 60.81% accuracy. Feature selection usually improved accuracy for all but Random Forests. There is little difference in results from tie-breaking approaches.

In Table II, results when performing leave one volume out with 3D features are shown. Without any feature selection being employed, J48 achieves an accuracy of 67.57%. With an accuracy of 66.22%, libSVM has the next highest accuracy using wrapper feature selection. Feature selection improved the accuracy of SVM, but resulted in lower accuracy for J48. Random Forests showed the highest accuracy of 60.81% with Relief-F feature selection using a subset of 25 features. IB5 achieves the same accuracy of 51.35% in each case, even

though the misclassification of the minority class was observed to improve with feature selection.

TABLE I
PERFORMANCE OF CLASSIFIER MODELS ON 2D FEATURES PERFORMING LEAVE-ONE-VOLUME-OUT (The highest accuracy for each classifier is in bold)

Classifier	Feature Selection	Accuracy (slice)	Accuracy ($Volume^1$)	Accuracy ($Volume^2$)
J48	None	49.32	45.94	47.30
	Relief-F (50 features)	47.63	47.30	48.65
	Relief-F (25 features)	47.47	43.24	45.95
	Wrapper	56.92	56.76	**56.76**
Random Forests	None	49.32	**52.70**	50.00
	Relief-F (50 features)	49.48	51.35	48.64
	Relief-F (25 features)	47.30	44.59	47.29
IB5	None	49.59	47.30	48.65
	Relief-F (50 features)	50.82	51.35	55.41
	Relief-F (25 features)	56.09	**58.11**	56.76
	Wrapper	53.27	56.76	56.76
libSVM	None	53.48	48.64	51.35
	Relief-F (50 features)	52.68	52.70	54.05
	Relief-F (25 features)	58.47	54.05	52.70
	Wrapper	53.29	59.46	***60.81***

TABLE II
PERFORMANCE OF CLASSIFIER MODELS FOR 3D FEATURES PERFORMING LEAVE-ONE-VOLUME-OUT (The highest accuracy for each classifier is in bold)

Classifier	Feature Selection	Accuracy (L-O-O)
J48	None	***67.57***
	Relief-F (50 features)	60.81
	Relief-F (25 features)	48.65
	Wrapper (forward selection)	45.95
Random Forests	None	59.46
	Relief-F (50 features)	54.05
	Relief-F (25 features)	**60.81**
IB5	None	51.35
	Relief-F (50 features)	51.35
	Relief-F (25 features)	51.35
	Wrapper (forward selection)	**51.35**
libSVM	None	52.70
	Relief-F (50 features)	56.76
	Relief-F (25 features)	56.76
	Wrapper (forward selection)	**66.22**

B. 10-fold Cross Validation

The 10-fold cross validation result is shown for 2D features in Table III and for 3D features in Table IV. In Table III, without feature selection being applied, libSVM had the best result with an accuracy of 56.76%. The highest accuracy of 62.16% was achieved with Random Forests with 50 features selected using Relief-F feature selection.

[1]Breaking Tie by choosing the class of the middle slice
[2]Breaking Tie by choosing the class of the largest tumor object

In Table IV the 3D 10-fold CV results are reasonably consistent with the evaluation for Leave One Out, with J-48 having the highest accuracy of 67.57%. Running classifiers without feature selection seems to yield better accuracy in the case of J48 and IB5, while feature selection improves the performance for Random Forests and SVM.

Using the Relief-F feature selection method to reduce the feature set to 50 features reduced accuracy for J48 (around 6% less accuracy) but improved SVM accuracy (around 7%). It had no effect on the performance of Random Forests, while for IB5 accuracy was vastly reduced (over 12% less accuracy) to 50%. When the feature set was further reduced to 25 features using Relief-F some increase in accuracy was obtained for Random Forests, while all other classifiers had lower accuracy. Wrapper forward selection did not have much of an effect in improving accuracy of the classifiers, except for SVM, which reached accuracy of 62.16%.

Table V shows percentage accuracy from performing a 5x2 fold Cross Validation on classifiers using 2D features, with the difference in accuracy from 3D features given within parentheses. The results indicate that using 2D features provides higher average accuracy over 3D features except for four instances, that have been highlighted in the table. Table VI shows the result of performing an F-test on results of the 5x2 Cross Validation on each classifier model, using 3D and 2D features respectively. An F-measure over 4.7351 indicates statistical significance over 95%. This level is not reached. In the cases where the 2D feature based classifiers had a higher F-value like J48 using wrappers, the overall accuracy is less than achieved with J48 and 3D features (albeit with no feature selection).

TABLE III
PERFORMANCE OF CLASSIFIER MODELS ON 2D FEATURES PERFORMING 10 FOLD CROSS VALIDATION (The highest accuracy for each classifier is in bold)

Classifier	Feature Selection	Accuracy ($Volume^1$)	Accuracy ($Volume^2$)
J48	None	**52.70**	48.65
	Relief-F (50 features)	45.95	47.30
	Relief-F (25 features)	47.30	48.65
	Wrapper	47.30	49.32
Random Forests	None	49.32	48.65
	Relief-F (50 features)	60.81	***62.16***
	Relief-F (25 features)	55.41	56.76
IB5	None	54.05	56.76
	Relief-F (50 features)	49.32	52.70
	Relief-F (25 features)	56.76	56.76
	Wrapper	**59.46**	58.11
libSVM	None	**56.76**	**56.76**
	Relief-F (50 features)	47.30	49.32
	Relief-F (25 features)	45.95	48.64
	Wrapper	49.32	48.64

TABLE IV
PERFORMANCE OF CLASSIFIER MODELS FOR 3D FEATURES PERFORMING 10 FOLD CROSS VALIDATION. (The highest accuracy for each classifier is in bold)

Classifier	Feature Selection	Accuracy (10-fold CV)
J48	None	***67.57***
	Relief-F (50 features)	60.81
	Relief-F (25 features)	52.70
	Wrapper (forward selection)	55.41
Random Forests	None	55.41
	Relief-F (50 features)	55.41
	Relief-F (25 features)	**58.11**
IB5	None	**62.16**
	Relief-F (50 features)	50.00
	Relief-F (25 features)	54.05
	Wrapper (forward selection)	45.96
libSVM	None	55.41
	Relief-F (50 features)	60.81
	Relief-F (25 features)	58.11
	Wrapper (forward selection)	**62.16**

TABLE V
PERFORMANCE OF CLASSIFIER MODELS PERFORMING 5X2 FOLD CROSS VALIDATION FOR 2D FEATURES, DIFFERENCE FROM 3D FEATURES GIVEN WITHIN PARENTHESES. (The case where 3D features perform better has been highlighted)

Classifier	Feature Selection	Accuracy (5x2 fold CV)
J48	None	**53.11 (-6.35)**
	Relief-F (50 features)	**51.75 (-0.14)**
	Relief-F (25 features)	**52.57 (-0.13)**
	Wrapper (forward selection)	53.92 (+3.16)
Random Forests	None	52.84 (+3.92)
	Relief-F (50 features)	52.30 (+1.76)
	Relief-F (25 features)	52.57 (+0.95)
IB5	None	52.16 (+3.51)
	Relief-F (50 features)	51.75 (+4.45)
	Relief-F (25 features)	50.00 (+1.90)
	Wrapper (forward selection)	53.65 (+5.26)
libSVM	None	**53.11 (-0.94)**
	Relief-F (50 features)	52.30 (+2.50)
	Relief-F (25 features)	49.73 (+0.81)
	Wrapper (forward selection)	50.00 (+1.61)

V. CONCLUSIONS

The feature selection approaches applied here were often ineffective for 3D features. It is likely that not all features are needed and some selection seems likely to improve the process. There was some improvement with 2D features. For support vector machines, finding the right parameter combination is essential, but our one attempt with 2D features showed no performance difference. Even though the number of 2D and 3D features is large, they are by no means exhaustive. While there are other features that might be of use, we believe this work generally covers the types of features needed to discriminate among lung tumor classes. Clinical features like tumor location within the lung and others can be taken into account along with image features to help uniquely identify tumors.

In this paper, we tried two different simple approaches to feature selection, and both have advantages and disadvantages. Relief-F ignores feature dependencies, but runs very fast. Relief-F was relatively ineffective in choosing a good set of features for this data. Wrappers model feature dependencies, however they run the risk of over fitting. The Wrapper forward selection approach was better in selecting features, but the stopping criterion stops it too early, we believe. We can use backward best first search, but search time then becomes a very significant cost. Clearly, a more sophisticated feature selection investigation is required as this preliminary investigation proceeds.

Tumors are certainly 3D, but feature selection could potentially improve performance in 2D. On the other hand, there are no ties to be broken with 3D features unlike the case where different slices of a tumor may be classified into separate classes. The most accurate classifiers often used 3D features. Feature selection improved accuracy more for 2D classifiers than 3D, however, it usually did not improve accuracy above that obtained by using all 3D features. There is no clear advantage in accuracy between 2D and 3D features, but 3D simplify classifier construction. Since a vast range of image features have been investigated here, it may be hypothesized that the two classes of tumors chosen for classification may require some specialized image features not developed here. In this first known attempt to do lung tumor identification from CT images, over 67% accuracy in identifying lung tumor type *just from CT images* was achieved, which is a good start.

TABLE VI
F-TEST ON 5X2 CROSS VALIDATION RESULTS BETWEEN 2D FEATURES AND 3D FEATURES

Classifier	None	Relief-F (50 features)	Relief-F (25 features)	Wrapper
J-48	3.89	2.43	3.79	3.18
Random Forests	3.25	2.43	2.08	X
IB5	2.43	2.20	1.73	2.63
libSVM	1.53	3.26	3.21	3.44

ACKNOWLEDGMENT

This work is partially supported by grant 1U01CA143062-01, Radiomics of NSCLC from the National Institutes of Health.

REFERENCES

[1] Mitchell E. Garber, Olga G. Troyanskaya, Karsten Schluens, Simone Petersen, Zsuzsanna Thaesler, Manuela Pacyna-Gengelbach, Matt van de Rijn, Glenn D. Rosen, Charles M. Perou, Richard I. Whyte, Russ B. Altman, Patrick O. Brown, David Botstein, and Iver Petersen: "Diversity of gene expression in adenocarcinoma of the lung", PNAS 2001 98: 13784-13789.

[2] I. Sluimer, A. Schilham, M. Prokop, and B. Ginneken: "Computer Analysis of Computed Tomography Scans of the Lung: A Survey" , IEEE Transactions on Medical Imaging, vol. 25, No. 4, 2006.

[3] S. Kido, K. Kuriyama, M. Higashiyama, T. Kasugai, and C. Kuroda, "Fractal analysis of internal and peripheral textures of small peripheral bronchogenic carcinomas in thin-section computed tomography: Comparison of bronchioloalveolar cell carcinomas with nonbronchioloalveolar cell carcinomas", J. Comput. Assist. Tomogr., vol. 27, no. 1, pp. 56 - 61, 2003.

[4] M. F. McNitt-Gray, N. Wyckoff, J. W. Sayre, J. G. Goldin, and D. R. Aberle, "The effects of co-occurrence matrix based texture parameters on the classification of solitary pulmonary nodules imaged on computed tomography", Comput. Med. Imag. Graphics, vol. 23, no. 6, pp. 339 - 348, 1999.

[5] Rene Korn, Arno Schaepe, Guenter Schmidt, Gerd Binnig and Claus Bendtsen: "Lung Tumor Analysis (LuTA) with Definiens Cognition Network Technology"

[6] Manduca A, Carston MJ, Heine JJ, Scott CG, Pankratz VS, Brandt KR, Sellers TA, Vachon CM, Cerhan JR: Texture features from mammographic images and risk of breast cancer. Cancer Epidemiol Biomarkers Prev 2009, 18:837-845.

[7] BB. M. The fractal geometry of nature. Freeman, San Francisco. 1983

[8] Goldberger AGAN-ldfcCt, fractals, and complexity at the bedside. Lancet 347:1312-1314. Non-linear dynamics for clinicians: Chaos theory, fractals, and complexity at the bedside. Lancet 347:1312-1314, 1996.

[9] Family F, Masters B, Platt D. Fractal pattern formation in human retinal vessels. Physica D: Nonlinear Phenomena 38:98, 1989.

[10] Smith TJ, Marks W, Lange Gea. A fractal analysis of cell images. J Neurosci Methods 27:173-180, 1998.

[11] Chellappa, R., Bagdazian, R.: Fourier Coding of Image Boundaries. IEEE Tans. Pattern Anal. Mach. Intell. 6(1) (1984) 102-105

[12] Mokji, M.M.; Abu Bakar, S.A.R.; , "Gray Level Co-Occurrence Matrix Computation Based On Haar Wavelet," Computer Graphics, Imaging and Visualisation, 2007. CGIV '07 , vol., no., pp.273-279, 14-17 Aug. 2007

[13] V.A. Kovalev, F. Kruggel, H.-J Gertz, and D.Y. von Cramon. "Three-dimensional texture analysis of MRI brain datasets." IEEE Trans. on Medical Imaging, 20(5):424-433,2001.

[14] D.-H Xu, A.S. Kurani, J.D. Furst, and D.S. Raicu. "Run-length encoding for volumetric texture." 4th IASTED Int'l Conf on Visualization, Imaging and Image Processing, 2004.

[15] Xiaoou Tang, "Texture information in run-length matrices," Image Processing, IEEE Transactions on , vol.7, no.11, pp.1602-1609, Nov 1998

[16] Laws K. Textured image segmentation [dissertation]. Los Angeles (CA): University of Southern California; 1980.

[17] Benke K K, Cox D and Skinner D R, "A study of the effect of image quality on texture energy measures", Meas. Sci. Technol. 5 400-7, 1994

[18] A.S. Kurani, D.-H Xu, J.D. Furst, and D.S. Raicu. "Co-occurrence matrices for volumetric data." 7th IASTED Int'l Conf on Computer Graphics and Imaging, 2004.

[19] K. Jafari-Khouzani, H. Soltanian-Zadeh, K. Elisevich, and S. Patel. "Comparison of 2D and 3D wavelet features for the lateralization." In Proc. of SPIE Medical Imaging 2004: Physiology, Function and Structure from Medical Images, volume 5369, pages 593-601, 2004.

[20] Quinlan, J.R.; , "Decision trees and decision-making," Systems, Man and Cybernetics, IEEE Transactions on , vol.20, no.2, pp.339-346, Mar/Apr 1990

[21] Breiman, L., Friedman, J., Olshen, R. and Stone, C. (1984). Classification and Regression Trees. Wadsworth, Belmont, CA.

[22] D.W. Aha, D. Kibler and M.K. Albert, Instance-based learning algorithms. Machine Learning 6 (1991), pp. 37-66.

[23] Hearst, M.A.; Dumais, S.T.; Osman, E.; Platt, J.; Scholkopf, B.; , "Support vector machines," Intelligent Systems and their Applications, IEEE , vol.13, no.4, pp.18-28, Jul/Aug 1998

[24] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann and Ian H. Witten: " The WEKA Data Mining Software: An Update", SIGKDD Explorations, Volume 11, Issue 1, 2009.

[25] Chih-Chung Chang and Chih-Jen Lin;, "LIBSVM : A library for support vector machines, 2001." Software available at http://www.csie.ntu.edu.tw/c̃jlin/libsvm

[26] I. Kononenko and L. De Raedt, Estimating attributes: analysis and extensions of Relief. In: F. Bergadano, Editor, Proceedings European Conference on Machine Learning (1994).

[27] Ron Kohavi, George H. John, Wrappers for feature subset selection, Artificial Intelligence, Volume 97, Issues 1-2, Relevance, December 1997, Pages 273-324, ISSN 0004-3702.

[28] J. Ross Quinlan, C4.5: programs for machine learning, Morgan Kaufmann Publishers Inc., San Francisco, CA, 1993

[29] C. J. C. Burges. "A Tutorial on Support Vector Machines for Pattern Recognition." Knowledge Discovery and Data Mining, vol.2, no.2, pp.121-167, 1998.

[30] K. Kramer, L. Hall, D. Goldgof, Fast Support Vector Machines for Continuous Data." IEEE Transactions on Systems, Man and Cybernetics, vol.39, no. 4, pp.989 - 1001, Aug 2009

[31] Dehmeshki, J.; Chen, J.; Casique, M.V.; Karakoy, M.; , "Classification of Lung Data by Sampling and Support Vector Machine," Engineering in Medicine and Biology Society, 2004. IEMBS '04. 26th Annual International Conference of the IEEE , vol.2, no., pp. 3194- 3197, 1-5 Sept. 2004

[32] Ethem Alpaydin, Combined 5x2 cv F Test for Comparing Supervised Classification Learning Algorithms, Neural Computation 11, 1885-1892 (1999)