Predictive modeling using shape statistics for interpretable and robust quality assurance of automated contours in radiation treatment planning

Zachary T. Wooten
Department of Statistics, Rice University, 6100 Main St.
Houston, TX 77005, USA
Email: ztw5@rice.edu

Cenji Yu and Laurence E. Court
Department of Radiation Physics, The University of Texas MD Anderson Cancer Center,
1400 Pressler St. Houston, TX 77030, USA
Email: cyu4@mdanderson.org, lecourt@mdanderson.org

Christine B. Peterson
Department of Biostatistics, The University of Texas MD Anderson Cancer Center,
1400 Pressler St. Houston, TX 77030, USA
Email: cbpeterson@mdanderson.org

Deep learning methods for image segmentation and contouring are gaining prominence as an automated approach for delineating anatomical structures in medical images during radiation treatment planning. These contours are used to guide radiotherapy treatment planning, so it is important that contouring errors are flagged before they are used for planning. This creates a need for effective quality assurance methods to enable the clinical use of automated contours in radiotherapy. We propose a novel method for contour quality assurance that requires only shape features, making it independent of the platform used to obtain the images. Our method uses a random forest classifier to identify low-quality contours. On a dataset of 312 kidney contours, our method achieved a cross-validated area under the curve of 0.937 in identifying unacceptable contours. We applied our method to an unlabeled validation dataset of 36 kidney contours. We flagged 6 contours which were then reviewed by a cervix contour specialist, who found that 4 of the 6 contours contained errors. We used Shapley values to characterize the specific shape features that contributed to each contour being flagged, providing a starting point for characterizing the source of the contouring error. These promising results suggest our method is feasible for quality assurance of automated radiotherapy contours.

Keywords: Shape statistics; Contour quality assurance; Medical imaging; Random forest.

1. Introduction

Segmenting anatomical structures in medical images is a critical step in radiation treatment planning, as treatment plans are optimized to achieve a high radiation dose to tumor while sparing nearby organs at risk. Recently, increasing effort has been put into automating the contouring process, as this would save clinicians time, reduce human error, and enhance access to radiation therapy in low-resource environments [1]. Deep learning methods like convolutional neural networks (CNN) have revolutionized the automation of contouring. While the results from these

methods are promising, they provide no measures to indicate uncertainty or low confidence in challenging cases. Deep learning methods can make mistakes in image segmentation and contouring, particularly when faced with real data that do not resemble instances in their training data. It is of critical importance to avoid contouring errors in radiotherapy planning, as contouring mistakes could lead to overdosage of organs at risk. Currently, automatically generated contours must be manually reviewed for errors. Creating an automated contour review process to find and flag problematic contours would be a more objective and efficient approach.

Some approaches have been proposed to tackle this challenge. McIntosh et al. (2013) used a groupwise conditional random forest to detect contour errors based on imaging features [2], while Hui et al. (2018) showed that volumetric features of a set of contours can be used to fit univariate parametric distributions and find outliers on each feature [3]. Rhee et al. (2019) showed promising results using a second CNN-based model for flagging unacceptable contours [4]. However, relying on a similar approach for contouring and quality assurance may create redundancy, as similar methods may fail in similar ways.

We propose an orthogonal method for flagging unacceptable contours that only uses shape features of the contour without relying on deep learning methods or image features. This approach was chosen to allow our method to be applicable across various imaging systems, as image intensity and radiomic features depend heavily on the platform used for image acquisition. Our method accurately flags erroneous contours based on aspects of the resulting shapes, avoiding dependence on the imaging modality. Specifically, we trained a random forest classifier on shape features of kidney contours and compared its performance to alternative machine learning methods in correctly flagging unacceptable contours. We demonstrate its application to an external data set, where we identify potential contouring errors and characterize the shape features that informed these predictions.

2. Background

2.1 Shape features

Shape features are quantitative summaries that aim to characterize the geometric aspects of an object. Existing works on shape analysis, including Dryden [5] and Wirth [6], provide numerous examples of shape features that can be used to describe various geometric properties. Here, we rely on the features listed in Table 1.

Since several of these shape features require computing the convex hull of an object, we provide some additional discussion of the convex hull and its properties. The convex hull of an object is the smallest convex shape that contains the object, as illustrated in Figure 1a. The area is the shaded portion, while the convex area is the portion within the convex hull, shown as a dotted outline. Furthermore, the perimeter of the shape is calculated from the outline of the shaded object, whereas the convex perimeter is calculated from the outline of the convex hull.

Additional features of interest include sphericity, which describes how closely the shape resembles a sphere (or circle in two dimensions) and is a ratio of the minimum radius to the

maximum radius. Naturally, for a circle, the minimum and maximum radii are the same. Hence the farther this ratio deviates from 1, the less circular the shape. Figure 1b illustrates how the minimum and maximum radii used in computing this shape statistic would be calculated.

Table 1. Shape features and their descriptions

Shape Feature	Description	Formula
Area	Number of pixels/voxels in a shape	
Perimeter	Length of number of pixels/voxels in the boundary of the object	
Minimum Radius	Shortest radius value from the center of shape to boundary	
Mean Radius	Average radius value from the center of shape to boundary	
Max Radius	Largest radius value from the center of shape to boundary	
Centroid Size	Square root of the sum of squared Euclidean distances from each landmark to the centroid [5]	$\sqrt{\sum_{i=1}^{k} \|\|(X)_i - \bar{X}\|\|^2}$
Compactness	The ratio of the area of an object to the area of a circle with the same perimeter	$\frac{4\pi * \text{Area}}{(\text{Perimeter})^2}$
Sphericity	The degree to which an object approaches the shape of a sphere	$\frac{\text{Min Radius}}{\text{Max Radius}}$
Convexity	The relative amount that an object differs from a convex object	$\frac{\text{Convex Perimeter}}{\text{Perimeter}}$
Solidity	The ratio of the area of an object to the area of a convex hull of the object	$\frac{\text{Area}}{\text{Convex Area}}$
Roundness	The ratio of the area of an object to the area of a circle with the same convex perimeter	$\frac{4\pi * \text{Area}}{(\text{Convex Perimeter})^2}$

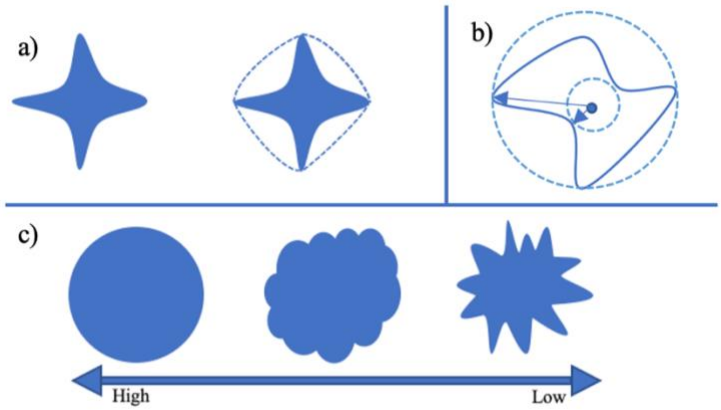

Fig 1. a) Shape with convex hull; b) Sphericity is the ratio of a shape's minimum and maximum radii; c) Shapes decreasing in value from left to right for compactness, convexity, solidity, and roundness.

Finally, we include the shape features compactness, convexity, solidity, and roundness. These four shape features take values from 0 to 1, where a higher value indicates the shape is smoother and less spiky than lower values. In Figure 1c, we see the circle on the left would have the highest value on these four shape statistics, and the irregular shape on the right would have the lowest value.

3. Methods

3.1 Training dataset

Our training data was obtained from CT scans for cervix radiotherapy treatment planning. Here we focus on contouring of the kidney; since most patients have two kidneys, this yields two structures per patient plan. The contours were generated by the Radiation Plan Assistant (RPA) [7], using a deep learning model based on a CNN algorithm. In total, we obtained 260 clinically acceptable contours using the RPA. A dosimetrist then manually created erroneous contours of several of the same kidney structures, yielding 52 unacceptable contours. Figure 2 provides an illustrative example showing acceptable and unacceptable contours of a patient's kidney. Typically, an organ at risk will be reflected in multiple image slices, where each slice captures a view of the patient's anatomy for a given orientation and depth.

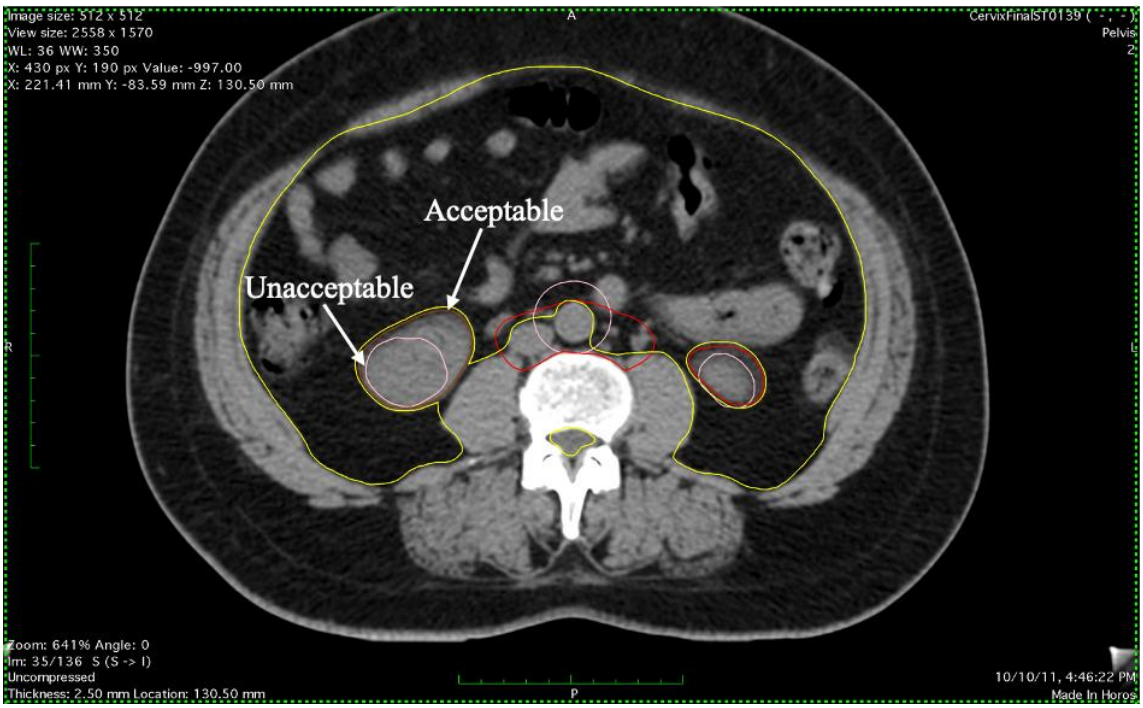

Fig. 2. An axial view of a cervix radiation treatment plan with organ structures contoured

To extract the contour for downstream analysis, we created a mask for the organ on a 512 by 512 voxel grid. The entries in the corresponding binary matrix representation were set to 1 if the voxel coordinate was contained within the contour boundary, and otherwise set to 0. We repeated

this for every axial slice in the plan until we had a complete three-dimensional array of the organ structure. The dimension of each voxel was 1.27mm x 1.27 mm x 2.5 mm.

3.2 Extracting shape features

We now describe how the shape features described analytically in Table 1 were computed in practice. We extracted shape features from the contours using R by inputting the binary matrix representation of the contour mask into various functions. The functions assume there is a single, closed contour. The perimeter and compactness of a contour were calculated by counting the number of voxels on its edge. We relied on the *EBImage* package to calculate the minimum, mean, and maximum radii, by finding the midpoint of the contour and the radii to each edge voxel [8]. With the radii values we calculated sphericity. We calculated the convex hull of a contour using the *chull* function in the *grDevices* package that returns coordinates of the convex hull [9]. We calculated the area and convex area of a contour using the *concaveman* package [10]. Finally, we relied on the *shapes* package to calculate the centroid size [11]. We captured these shape features for every slice in the patient's radiotherapy treatment plan, resulting in a vector of values for each feature across slices.

3.3 Histogram and volumetric features

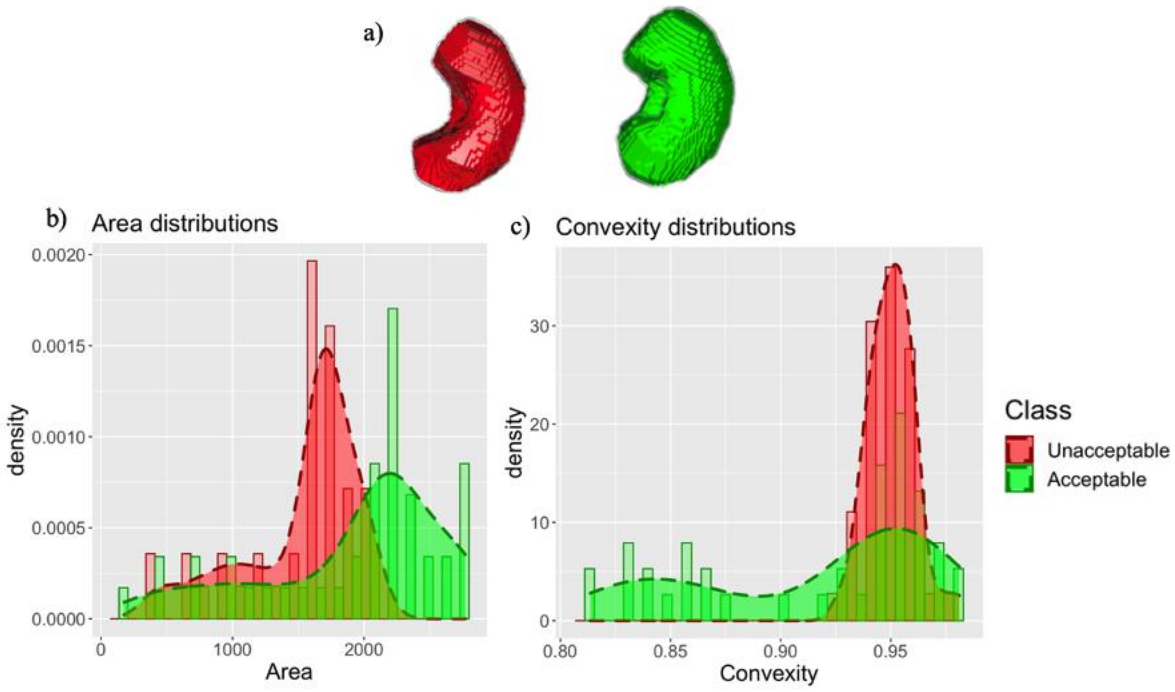

Fig. 3. a) 3D rendering of the unacceptable (red) and acceptable (green) contours of the right kidney; b) distributions of the areas; c) distributions of convexity.

A challenge in treating these shape features as predictors in a model is that the organ structures vary in size across patients, resulting in vectors of different lengths. For example, some structures could be defined in 50 slices, while others could be defined in 100 slices. In addition, values from neighboring slices tend to be highly correlated. To construct a consistent set of summary features, we relied on histogram features which summarize the distribution of shape values for each organ.

Specifically, we take all the values from a specific shape feature, and we calculate the minimum, 1st quartile, median, mean, 3rd quartile, maximum, and standard deviation. Figure 3 illustrates an unacceptable and an acceptable 3D structure, along with the shape feature distributions for area and convexity. Here, we can see a distinct difference in the shape feature distributions. We augmented our feature set by including volume, surface area, and the volume to surface area ratio. This resulted in a total of 80 features per structure.

3.4 Machine learning classifier

3.4.1 The random forest algorithm

Random forests are a popular machine learning algorithm that use an ensemble of decision trees [12]. Each tree casts a vote for the most popular class per input vector. The trees in the random forest are created by partitioning the feature space into rectangular regions on a randomly chosen set of features called nodes. Based on an optimization criterion, the tree splits at a particular value in the feature space. The decision trees created are "weak learners," meaning a single tree alone would have poor accuracy in classification. However, together the trees break up the feature space uniquely and make powerful predictions. Random forests are robust to challenging settings, and can accommodate non-linear effects, interactions among features, and correlated predictors. In addition to strong predictive performance, random forests can provide insight on the relative importance of predictors through variable importance scores. To develop our random forest model, we used the *randomForest* package in R with 500 trees and 16 node splits per tree.

3.4.2 Comparators

To assess the performance of the random forest relative to that of other machine learning approaches, we applied other popular classifiers including logistic regression, lasso logistic regression [13], naïve Bayes [14], and extreme gradient boosting (XGBoost) [15].

3.4.3 Model training and performance metrics

To train the classifiers, we performed repeated 5-fold cross validation on all 312 kidney observations. For each fold we used roughly 80% of the data as a training set and 20% of the data as a test set. Performance metrics including the area under the curve (AUC) for the receiver operating characteristic (ROC) and precision-recall (PR) curves were computed on each test set and averaged over folds and replicates. We also computed the sensitivity and specificity using a default threshold value of 0.5 and an optimized threshold obtained using Youden's Index.

3.4.4 Shapley values

In a machine learning framework, the Shapley value can be used to explain model predictions by calculating each feature's contribution in a particular instance [16]. The contribution for a given feature is calculated by removing that feature from the model and seeing how the prediction value changes. If removing a feature drastically changes the prediction, then that feature would have a large Shapley value. Importantly, unlike variable importance scores, which provide a single ranking of features for the entire data set, Shapley values are case-specific. Using the *shapr* package in R, we applied this framework to identify key features driving the model predictions [17]. The resulting Shapley values were plotted as a bar chart to provide a starting point for identifying why specific contours were flagged.

4. Results

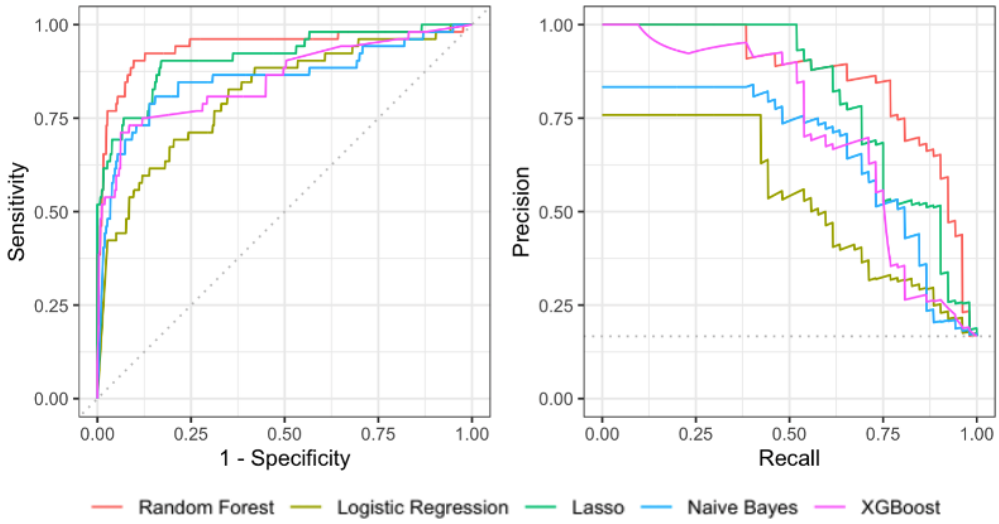

Fig. 4. ROC and PR Curves of various classifiers

In Table 2, we provide a summary of predictive performance in terms of the AUC for the ROC and PR curves, sensitivity and specificity using a threshold of 0.50, and sensitivity and specificity using an optimized threshold from Youden's index (indicated by subscripts). The metrics in Table 2 reflect

Table 2. Performance metrics from 10 iterations of five-fold cross validation

Classifier	Random Forest	Logistic Regression	Lasso	Naïve Bayes	XGBoost
AUC_{roc}	0.937 (± 0.008)	0.809 (± 0.013)	0.912 (± 0.009)	0.849 (± 0.008)	0.831 (± 0.020)
AUC_{pr}	0.828 (± 0.022)	0.506 (± 0.033)	0.829 (± 0.011)	0.647 (± 0.018)	0.655 (± 0.067)
$Specificity_{0.50}$	0.977 (± 0.005)	0.861 (± 0.014)	0.271 (± 0.019)	0.920 (± 0.004)	0.970 (± 0.011)
$Sensitivity_{0.50}$	0.608 (± 0.016)	0.640 (± 0.060)	0.983 (± 0.014)	0.692 (± 0.013)	0.571 (± 0.044)
$Specificity_{yi}$	0.883 (± 0.042)	0.817 (± 0.072)	0.902 (± 0.057)	0.878 (± 0.030)	0.879 (± 0.101)
$Sensitivity_{yi}$	0.889 (± 0.053)	0.733 (± 0.076)	0.808 (± 0.043)	0.816 (± 0.062)	0.719 (± 0.103)

averages over 10 replicates of five-fold CV. The AUC for the ROC curve summarizes predictive performance in terms of sensitivity and specificity across a range of threshold values. The PR curve is like the ROC curve but focuses on the trade-off between precision (also known as positive predictive value) and recall (also known as sensitivity). The PR curve is particularly useful in characterizing classification accuracy for imbalanced data sets. The proposed random forest prediction model outperformed the other classifiers with a cross-validated AUC_{roc} value of 0.937 and one of the highest AUC_{pr} value of 0.828 (similar to the value achieved by lasso logistic regression). Figure 4 shows illustrative ROC and PR curves from one replicate of the five-fold CV. In Table 2, we also provide sensitivity and specificity for specific cut-off values, where an instance is considered as flagged if its predicted value is above the threshold. We considered 0.50 as a standard cut-off and an optimized cut-off obtained using Youden's Index. In the radiation therapy quality assurance setting, a more sensitive classifier is preferred to ensure that concerning cases will get additional review. The random forest with Youden's index performed very well in this regard, achieving a sensitivity of 0.889. To illustrate, figure 5 shows the probabilities of each contour from the random forest trained on the entire dataset. Contours with probabilities above the threshold values are flagged as unacceptable. Shape features and code to reproduce analysis provided at: https://github.com/wootz101/QA_Contours

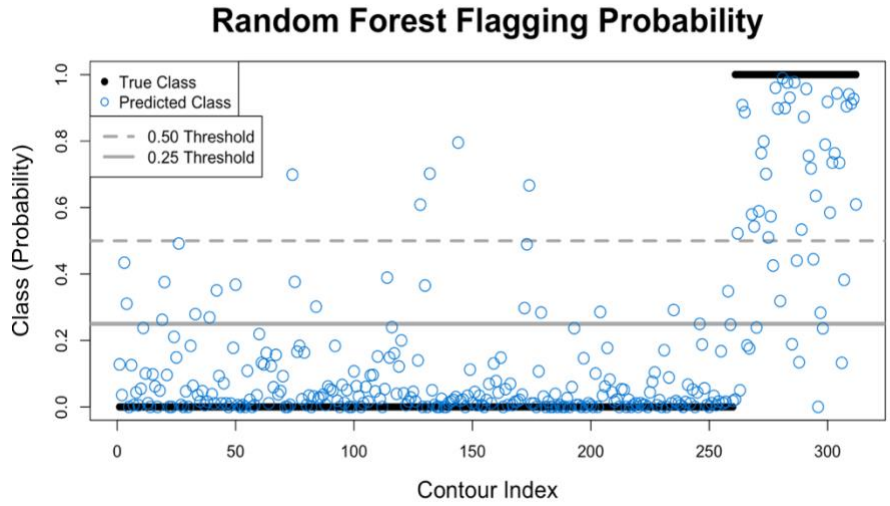

Fig 5. Random forest probabilities in blue with example thresholds in grey; the true class is marked in black, where acceptable contours have a value of 0 and unacceptable contours have a value of 1. The index range of 1-260 correspond to acceptable contours and 261-312 correspond to unacceptable contours.

5. Application to unlabeled data

Table 3. Error Rates

Model	Ground Truth	Not Flagged	Flagged	Class Error
80 Variable	Acceptable	255	5	1.9%
	Unacceptable	16	36	30.8%
Top 10 Variable	Acceptable	250	10	3.8%
	Unacceptable	15	37	28.8%

Based on these results, the random forest prediction method performed well at discerning acceptable vs. unacceptable contours in a cross-validation setting. We then sought to assess the utility of this approach when applied to a new external data set. To do so, we first trained a final random forest model using the entire dataset of 312 kidney contours, using the same parameters as before. Training on the full dataset, the random forest performs well with a total accuracy of 93.27% and an AUC value of 0.937, with a false positive rate of 1.9% and a false negative rate of 30.8%. Table 3 gives further information on the random forest's error rates based on a 50% threshold.

5.1 Variable importance

The random forest is a useful classifier in this regard as it also provides a measure of feature importance. Table 4 shows the top ten variables of importance by their inclusion mean decrease in accuracy percent.

Table 4. Importance measure

1st	2nd	3rd	4th	5th
Sphericity (Max) 2.7%	Min Radius (Min) 1.6%	Centroid (SD) 1.2%	Min Radius (SD) 1.2%	Area (SD) 1.1%
6th	7th	8th	9th	10th
Perimeter (SD) 1.1%	Mean Radius (SD) 0.9%	Max Radius (Median) 0.7%	Area (Min) 0.6%	Solidity (Mean) 0.6%

The *shapr* package in R is limited to 13 variables as the computation time increases exponentially with the number of variables. Therefore, we constructed a new random forest that only uses these top 10 shape histogram features to accommodate the software and hardware constraints. We used 500 trees and 8 node splits per tree as parameters. We lowered the node splits from 16 to 8 because we went from 80 to 10 input features. Trimming down the original model is an important step in order to use Shapely values to interpret why a contour gets flagged. Table 3 shows the performance of the random forest when we scale down from 80 features to the top 10. These results indicate the top 10 variable random forest model performs similarly to the full 80 variable model. In fact, the top 10 model is slightly more sensitive in flagging unacceptable contours.

5.2 Unlabeled dataset

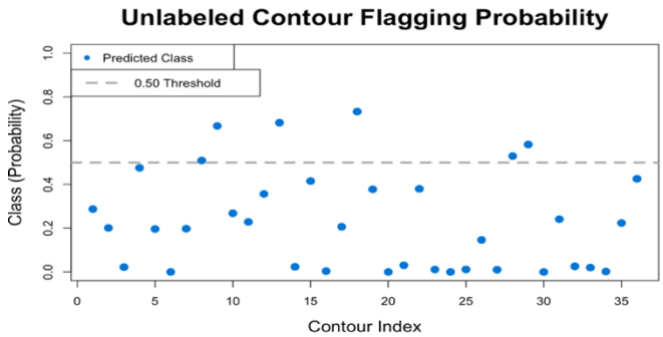

Fig 6. Probability of unacceptable contours from unlabeled dataset

We obtained an external data set of 18 radiation treatment plans for cervical cancer radiotherapy. The voxel dimensions of these plans were 1.172 mm x 1.172 mm x 2.5 mm. From these plans, we extracted 36 kidney contours. These independent test contours were previously unseen and so were considered unlabeled data. We extracted the shape features as previously described and applied our trained random forest to obtain model predictions. Figure 6 shows the estimated probabilities of each contour being unacceptable for use in radiotherapy planning. A total of 6 contours were flagged with a probability > 0.5.

5.3 Shapley values of flagged contours

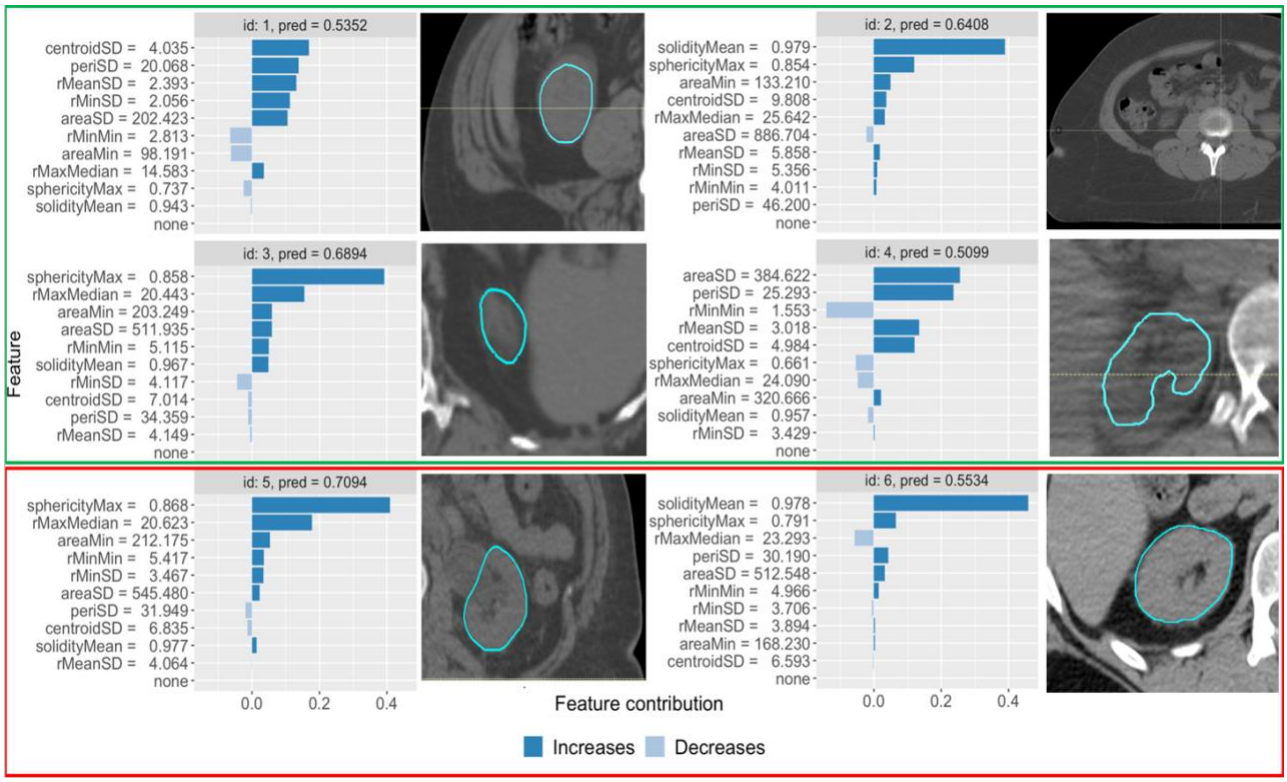

Fig 7. Shapley values show the impact each feature has on the overall prediction for the corresponding contour, with dark blue increasing and light blue decreasing the prediction of an error. The id: 1-4 are correctly flagged and outlined in green, and id: 5-6 are incorrectly flagged and outlined in red.

As would happen in the potential clinical application of our approach, an expert reviewer then inspected the flagged contours to simulate the clinical workflow. Of the 6 contours flagged, 4 were found to contain errors including over-contouring and under-contouring of the kidney region. Figure 7 shows the Shapley values of each variable for the flagged contours along with example images of the unacceptable kidney contours that were correctly flagged and the acceptable kidney contours that were incorrectly flagged. The errors in these contours are visually noticeable, with under-contouring being the most common error. Using the Shapley values, we can interpret how the deep learning contour erred. For instance, examining the Shapley value plot and corresponding contour for id: 1, we see the random forest model flagged the contour because the contour's centroid size,

perimeter, mean radius, and minimum radius had low standard deviations. The generated contour was indeed under-contoured which explains its out of distribution metrics. For id: 2, the contour had minor errors as it didn't contour the beginning of the kidney which resulted in a large mean solidity value. Hence, we see there is no contour in the medical image for id: 2 where there should be one. We see in id: 3 the Shapley value plots indicate that the maximum sphericity value was too high. The kidney was over-contoured on this patient which led to a highly spherical shape that the random forest noticed and flagged. For id: 4 we see that the area standard deviation and perimeter standard deviation values for the contour were too low, causing it to be flagged. Low standard deviation of area and perimeter would indicate that the area and perimeter values varied less from slice to slice than they did for acceptable contours. This real data application highlights the feasibility of our approach for radiotherapy quality assurance.

Our method also has limitations and sometimes generates false positives. We see in id: 5 the contour was flagged due to its high maximum sphericity value, however, there were no contouring errors found. This false positive is particularly interesting because it has the highest prediction value for being flagged. False positives are to be expected due to the inherent variation in human anatomy; our expert reviewer noted that in this instance the kidney structure was completely connected to a neighboring structure. The connectedness of the structure might lead to some variation in contouring. While this contour is safe for clinical use, it is challenging for both humans and machines to distinguish the ground truth border for this patient. For id: 6 the solidity mean value was too high which caused the contour to be flagged even though there were no errors.

6. Discussion

We have shown that training a random forest on shape features of contours is a viable method of contour quality assurance. Our method is novel and would be robust to differences in imaging platform or imaging processing steps in that it only requires shape features, and no imaging or radiomic features. Classification of contours using shape features could be useful in other contexts beyond radiation treatment planning; in particular, segmentation of the brain is a key task in the analysis of MRI data, while automatic detection of objects in images is a critical step in the development of automated driving systems. In both cases, critical structures identified using deep learning or other automated tools could potentially be distinguishable using shape features.

One of the limitations in this study is that the unacceptable contours used in the training data were created by hand. Since only acceptable contours are used in clinical radiotherapy treatment planning, real-world cases of unacceptable contours are difficult to obtain. Our method provides basic annotations to characterize which features drove the model predictions. More detailed information, including the spatial locations with potential errors, would enhance the interpretation of results. We plan to explore methods to enable location-specific annotation within contours in future work. Furthermore, we plan to explore how this method performs across other imaging platforms.

7. Acknowledgements

ZTW was partially supported by NIH/NCI training grant T32 CA096520 and NSF GRFP Grant No. 1842494. CBP was partially supported by NIH/NCI CCSG P30 CA016672 (Biostatistics Resource Group) and by a grant from Varian Medical Systems.

8. Bibliography

[1] McCarroll RE, Beadle BM, Balter PA, et al. (2018) Retrospective validation and clinical implementation of automated contouring of organs at risk in the head and neck: a step toward automated radiation treatment planning for low-and middle-income countries. *J Glob Oncol*. 4:1-11.
[2] McIntosh C, Svistoun I, Purdie TG. (2013) Groupwise conditional random forests for automatic shape classification and contour quality assessment in radiotherapy planning. *IEEE Trans Med Imaging*. **32**(6):1043-1057.
[3] Hui CB, Nourzadeh H, Watkins WT, et al. (2018) Quality assurance tool for organ at risk delineation in radiation therapy using a parametric statistical approach. *Med Phys*. **45**(5):2089-2096.
[4] Rhee DJ, Cardenas CE, Elhalawani H, et al. (2019) Automatic detection of contouring errors using convolutional neural networks. *Med Phys*. **46**(11):5086-5097.
[5] Dryden IL, Mardia KV. (2016) *Statistical Shape Analysis with Applications in R*. Wiley, 2nd edition.
[6] Wirth MA. (2004) Shape Analysis & Measurement. University of Guelph. Accessed online at http://www.cyto.purdue.edu/cdroms/micro2/content/education/wirth10.pdf.
[7] Kisling K, McCarroll R, Zhang L, et al. (2018) Radiation planning assistant - a streamlined, fully automated radiotherapy treatment planning system. *J Visualized Exp*. 134:e57411.
[8] Pau G, Fuchs F, Sklyar O, Boutros M, and Huber W (2010): EBImage - an R package for image processing with applications to cellular phenotypes. *Bioinformatics*, **26**(7), pp. 979-981, 10.1093/bioinformatics/btq046
[9] R Core Team (2020). R: A language and environment for statistical computing. *R Foundation for Statistical Computing*, Vienna, Austria.
[10] Gombin J, Vaidyanathan R and Agafonkin V (2020). concaveman: A Very Fast 2D Concave Hull Algorithm. R package version 1.1.0.
[11] Dryden IL (2019). shapes: Statistical Shape Analysis. R package version 1.2.5.
[12] Breiman L. (2001) Random Forests. *Machine Learning*, **45**(1): 5-32.
[13] Friedman J, Hastie T, Tibshirani R (2010). "Regularization Paths for Generalized Linear Models via Coordinate Descent." *Journal of Statistical Software*, **33**(1), 1–22.
[14] Majka M (2019). _naivebayes: High Performance Implementation of the Naive Bayes Algorithm in R_. R package version 0.9.7
[15] Chen T, Guestrin C. XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016 Aug 13 (pp. 785-794).
[16] Cohen S, Ruppin E, Dror G. (2005) Feature selection based on the Shapley value. *In Other Words*, 665-670.
[17] Sellereite N, Jullum M and Redelmeier A (2021). shapr: Prediction Explanation with Dependence-Aware Shapley Values. R package version 0.2.0.