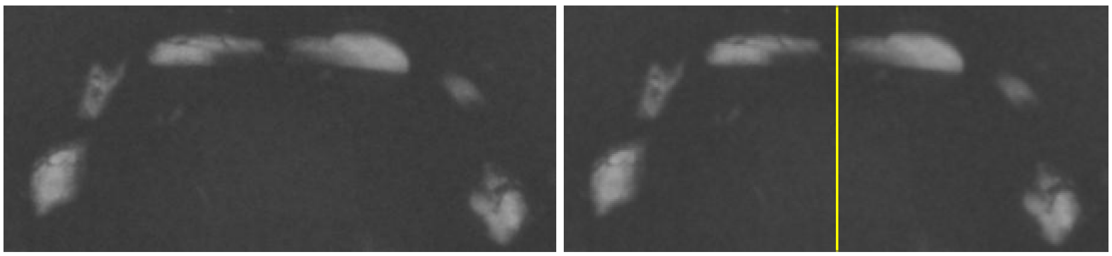
Feature Extraction from Dentition (Upper Jaw) Images

A. Data Format

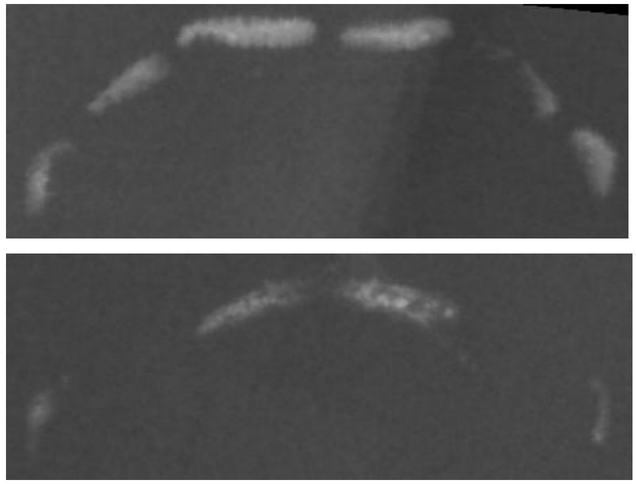


The Algorithm requires images (grayscale) to be in the above format (left), such that a line joining the left and the right incisor is parallel with X-axis and it should be cropped in such a way that there is a fixed margin between borders and key tooth prints (incisors and canines). This algorithm computes a structure by only considering incisor and canine tooth prints, hence any part below the canines is excluded by cropping as said previously.

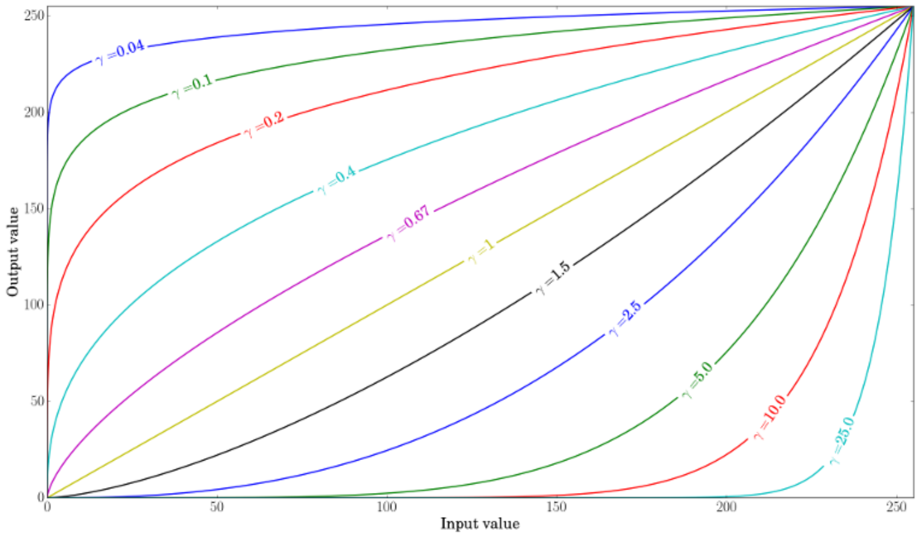
In addition, it's preferred that prints of key tooth prints are non-overlapping with neighboring prints meaning there must be sufficient space between vital tooth prints and irrelevant adjacent prints. However, left and right incisors will likely to be overlapped with each other. To solve this problem, an x-coordinate value (w.r.t. image plane) must be provided to separate them as shown in the image above (right). Such is required because splitting from the middle point sometimes cuts through the incisor prints.

For the algorithm to work on different image resolutions (of similar aspect ratio) a parameter to convert pixel length to millimeters must be provided. The same is also required to calculate various features. Fixing the scale to a specific metric (millimeters) that is independent of the image resolution is critical in designing such an algorithm.

B. Image Preprocessing



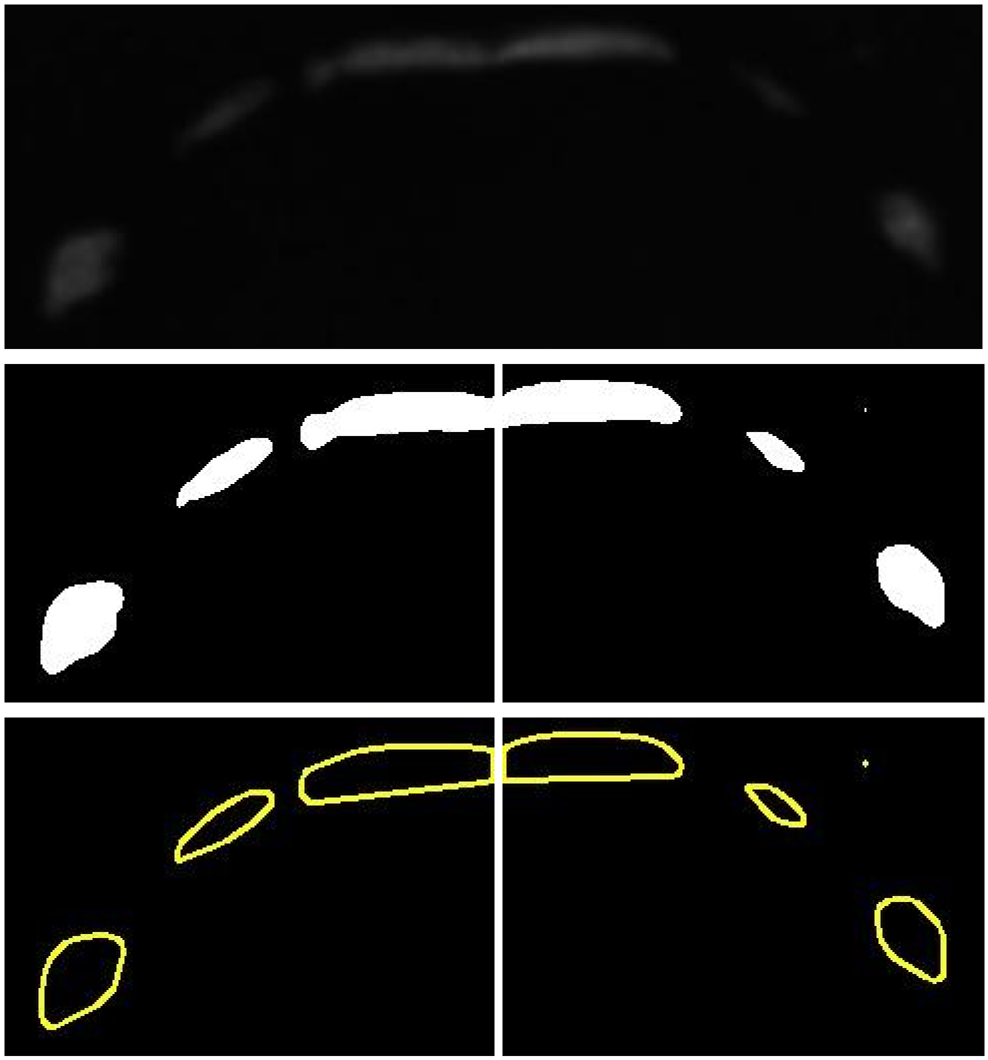
To accurately identify the center points of key tooth prints, the image must be binarized to exclude all the background while keeping all tooth prints. The above images show that directly applying thresholding to binarize the image clearly won’t work due to two reasons. Background pixels are sometimes as bright as lateral and canine tooth prints. Secondly, background pixels seem to be very noisy rather than constant.



This problem is solved by using the gamma correction. It maps pixel values non-linearly. Tooth prints are mostly very bright around their center and background pixels are not as bright as them. Gamma correction with the gamma value greater than 1 maps not so bright pixels to the significantly darker colors and does not do the same to very bright pixels (see above graph). Such only keeps tooth prints while remarkably suppressing the background. However, it does dim the key pixels. But it isn't a problem as long as all key regions are visible, even slightly. The result of gamma correction on an image with the value of gamma set to 3 is shown above. The value of gamma is an important hyperparameter.

As a result of gamma correction, regions containing tooth prints get shrunk by a small amount. Also, even for tooth prints, the pixels are not constantly bright throughout the region but it's rather noisy on the edges. To solve both of these issues, blurring is applied several times on a gamma-corrected image. Doing the same would enlarge the shrunk region, while also reducing the noise present at tooth print boundaries. The number of times to perform blurring is the second hyperparameter.

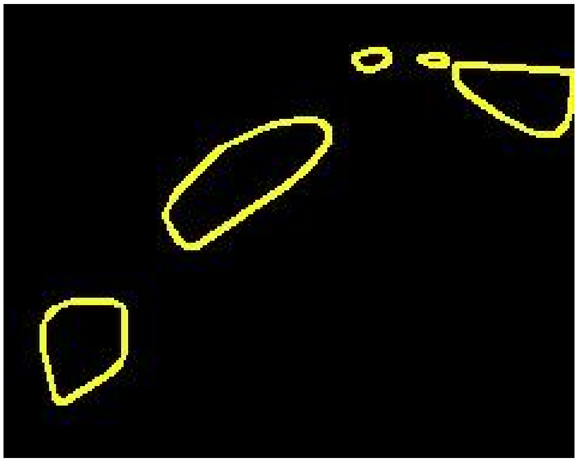
C. Contour Detection

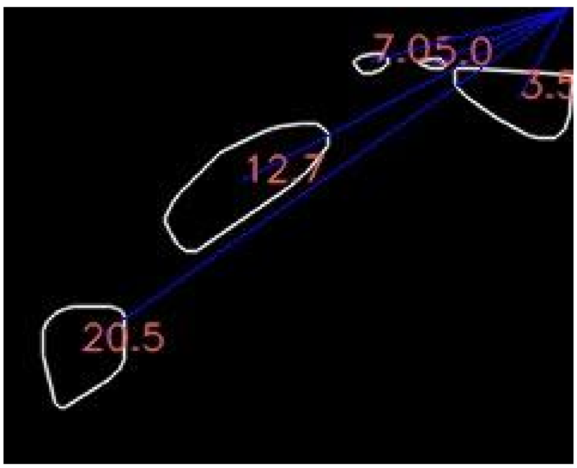


As a next step, the image is split into two parts provided the x-coordinate for the split as discussed previously. Both halves are processed separately from this step onwards. The threshold for binarizing is set to a low value due to the effect of gamma correction. Results of thresholding and contour (an enclosed shape) detection are shown above. Here, the threshold value is set to 10. It's a third vital hyperparameter.

D. Identification of Essential Tooth Prints

1. Left Incisor





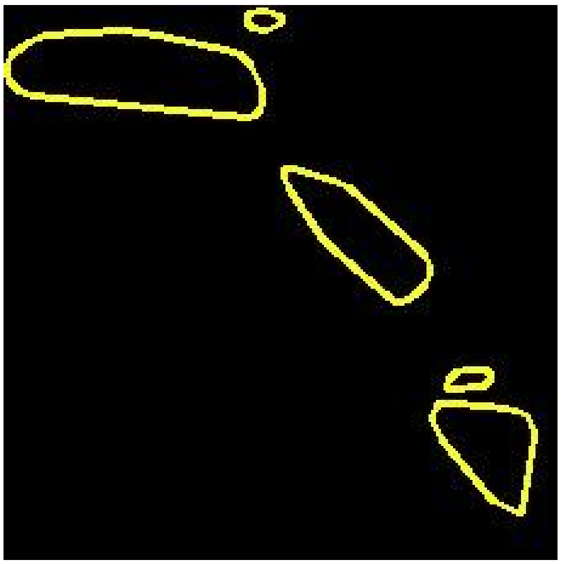
To find the left incisor given all the detected contours, it is observed that in any case centroid of the left incisor will always be closest to the top right corner of the left half image. The distance of lateral and canine will be much larger compared to the left incisor from the top right corner of the image.

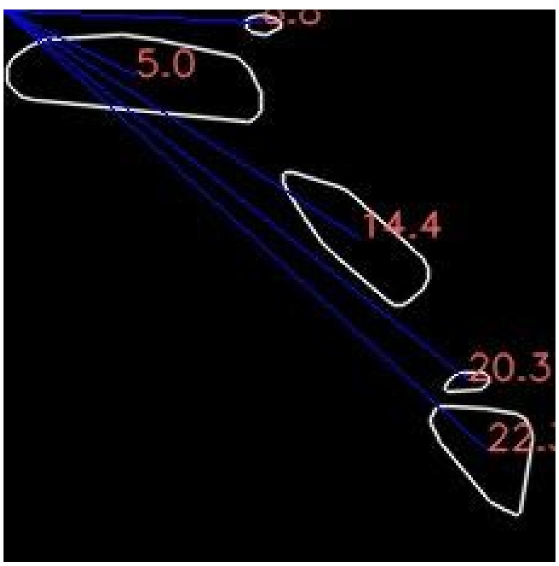
Here, the distance is measured in millimeters, pixel distance is converted to mm given the parameter (see Data Format section). All the contours whose distance from the top right corner is greater than the predefined threshold (one more hyperparameter) are discarded.

The left incisor is selected from the remaining contours, by picking the contour which has the highest area. The reason being sometimes the left incisor is divided into multiple contours rather than being one. Such a role will handle all such cases. Results are shown above, the contour on the top right with the distance of 3.5 mm is selected and others having lesser surface areas are ignored. This rule works well when the dominant contour represents a large portion of the left incisor and in case of other unimportant false positives which pass threshold criteria.

When our region of interest is segmented into multiple close-by contours rather than only one, this algorithm will pick the segment with the highest area. This would make an erroneous estimation of the location of the left incisor centroid because the true centroid will be at a different position.

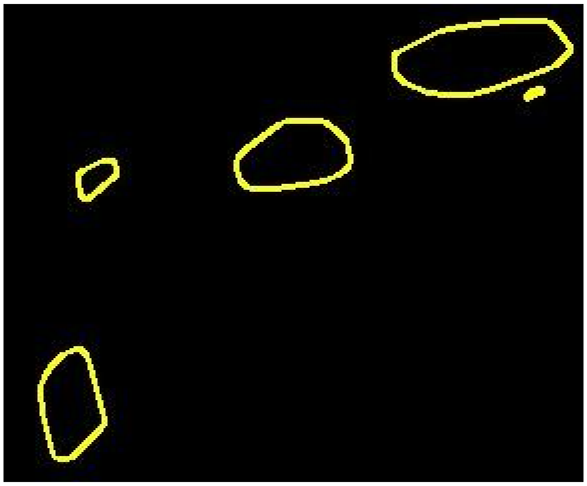
2. Right Incisor

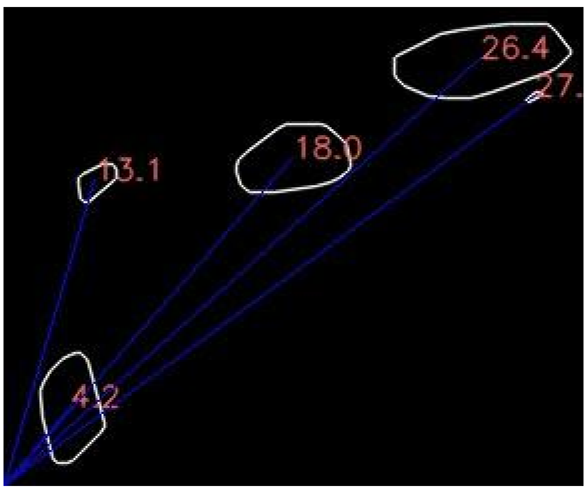




To find the right incisor given all the detected contours, it is observed that in any case centroid of the right incisor will always be closest to the top left corner of the right half image. Similar logic as previous is applied here as well to select the right incisor. Results are shown above, contour with a distance of 5 mm is picked.

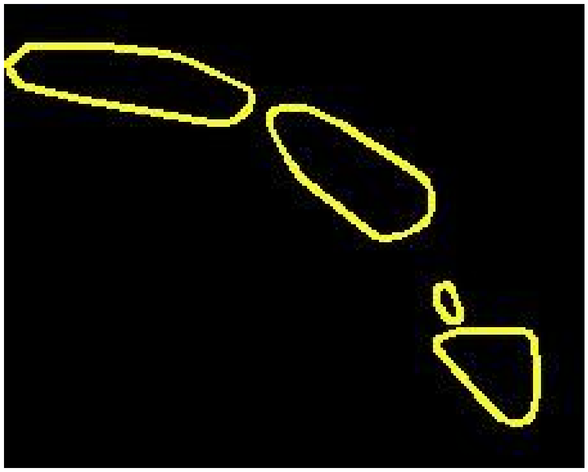
3. Left Canine

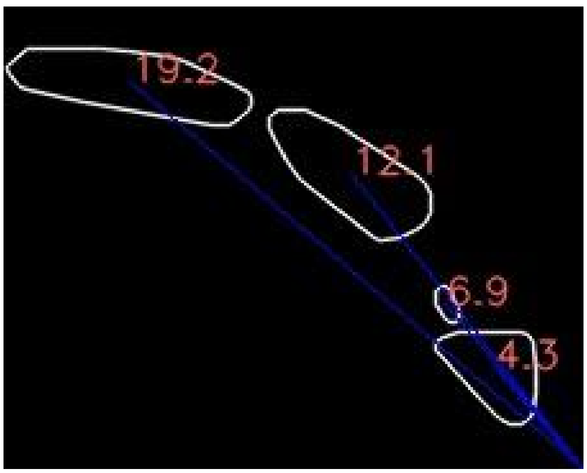




To find the left canine given all the detected contours, it is observed that in any case centroid of the left canine will always be closest to the bottom left corner of the left half image. Similar logic as previous is applied here as well to select the left canine. Results are shown above, contour with a distance of 4.2 mm is picked.

4. Right Canine



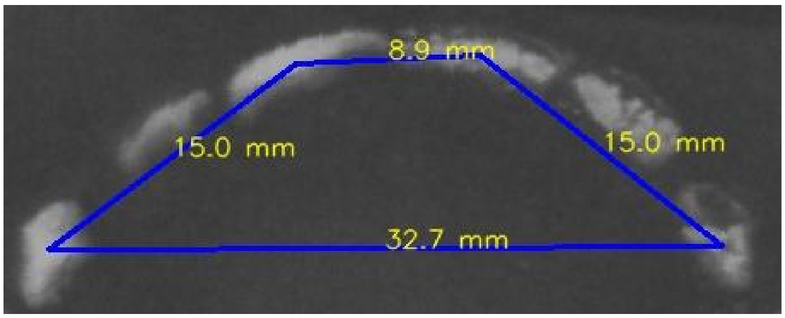


To find the right canine given all the detected contours, it is observed that in any case centroid of the right canine will always be closest to the bottom right corner of the right half image. Similar logic as previous is applied here as well to select the right canine. Results are shown above, contour with a distance of 4.3 mm is picked.

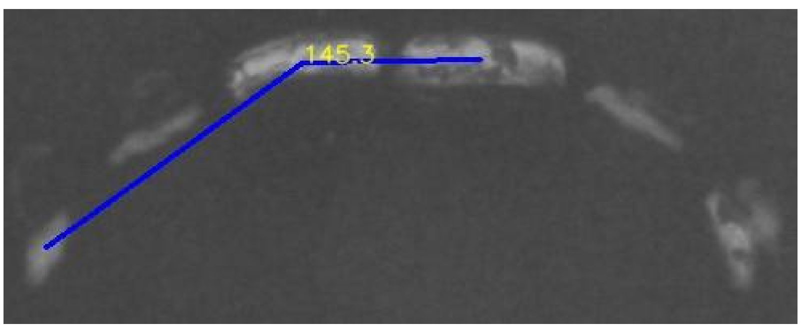
The process of identifying these 4 key locations introduced four more hyperparameters related to the distance threshold.

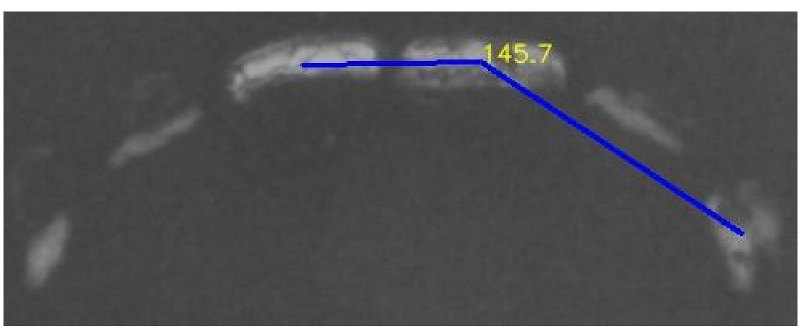
E. Feature Calculation

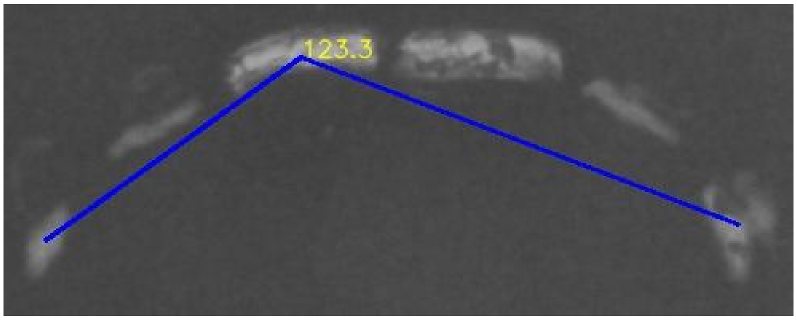
Once all four centroids are successfully detected relevant computed information is updated with respect to the whole image. The following features are then calculated from the extracted information,

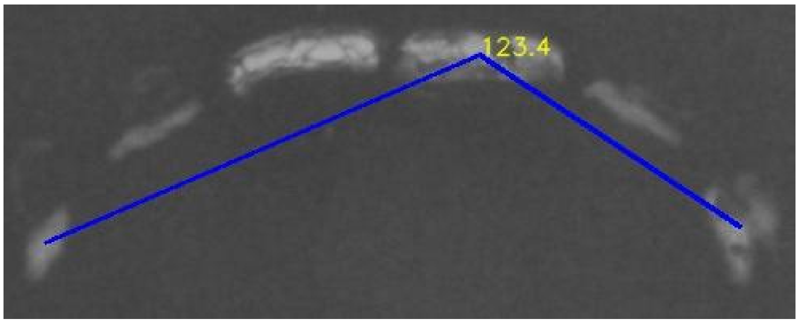


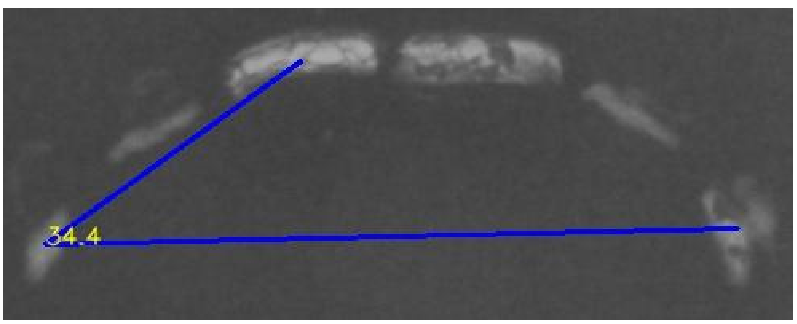
- Left Incisor - Right Incisor Distance (mm)
- Left Canine - Right Canine Distance (mm)
- Left Incisor - Left Canine Distance (mm)
- Right Incisor - Right Canine Distance (mm)

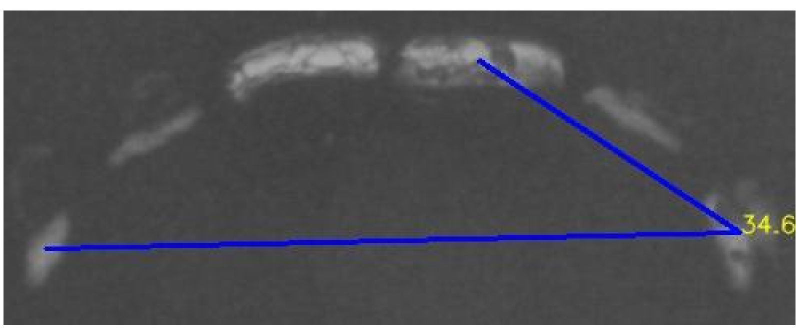








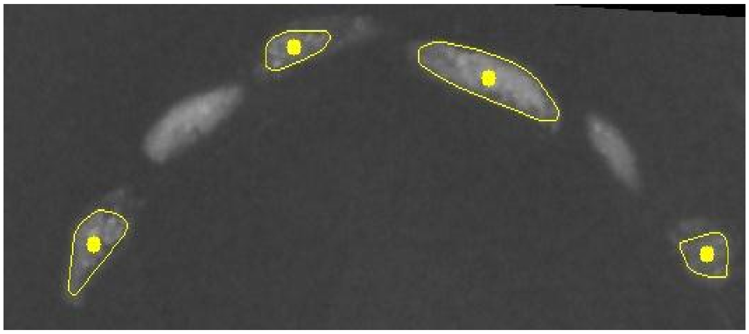


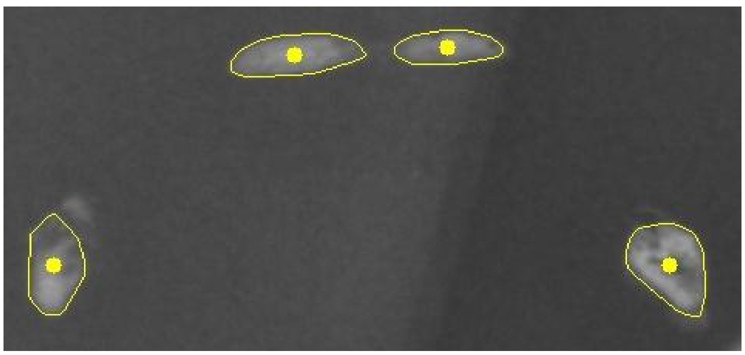


- LC - LI - RI Angle in degrees
- RC - RI - LI Angle in degrees
- LC - LI - RC Angle in degrees
- RC - RI - LC Angle in degrees
- LI - LC - RC Angle in degrees
- RI - RC - LC Angle in degrees

All these features might help uniquely identify the dentition of a person

F. Conclusion





The results of this algorithm are shown above. All discussed hyperparameters control how effectively and accurately the algorithm extracts features. Changing such would affect the performance and behavior of this process.

Their values can be tuned to work on a set of images picked from a specific data distribution called D1. The same values might not work on every single image, distribution of data matters a lot in this case. A significant shift in data distribution (different data distribution D2) would throw the algorithm into disarray, in such cases values of one or more hyperparameters must be tuned again to get desired results. However, other examples picked from the same distribution D1 can be easily worked upon accurately without hyperparameter re-tuning.

This phenomenon greatly impacts performance when it is put to work in the real world. To operate on real-world data with minimal human intervention and with high accuracy, major changes would be required in the current technique.

G. References

- A. Chaudhary and S. S. Singh, "Lung Cancer Detection on CT Images by Using Image Processing," 2012 International Conference on Computing Sciences, 2012, pp. 142-146, doi: 10.1109/ICCS.2012.43.
- A. M. Hasan, H. A. Jalab, F. Meziane, H. Kahtan, and A. S. Al-Ahmad, "Combining Deep and Handcrafted Image Features for MRI Brain Scan Classification," in IEEE Access, vol. 7, pp. 79959-79967, 2019, doi: 10.1109/ACCESS.2019.2922691.
- Jun-Ichiro Toriwaki, Yasuhito Suenaga, Toshio Negoro, Teruo Fukumura, "Pattern recognition of chest X-ray images", Computer Graphics and Image Processing, Volume 2, Issues 3–4, 1973, Pages 252-271, ISSN 0146-664X, https://doi.org/10.1016/0146-664X(73)90005-1.
- Putzu, Lorenzo, and Cecilia Di Ruberto. "White blood cells identification and counting from microscopic blood image." *Proceedings of world academy of science, engineering, and technology*. No. 73. World Academy of Science, Engineering, and Technology (WASET), 2013.