VISUAL INSPECTION OF BLADE TOOLS

PROBLEM STATEMENT

I was required to find the angles associated with teeth of a blade tool. I was given microscopic images of the blade tool and tasked to find all the angles associated with the teeth. Furthermore, the task required verification of the rectilinear profile of each tooth. This objective was achieved through a polygonal approximation function available in the OpenCV library.

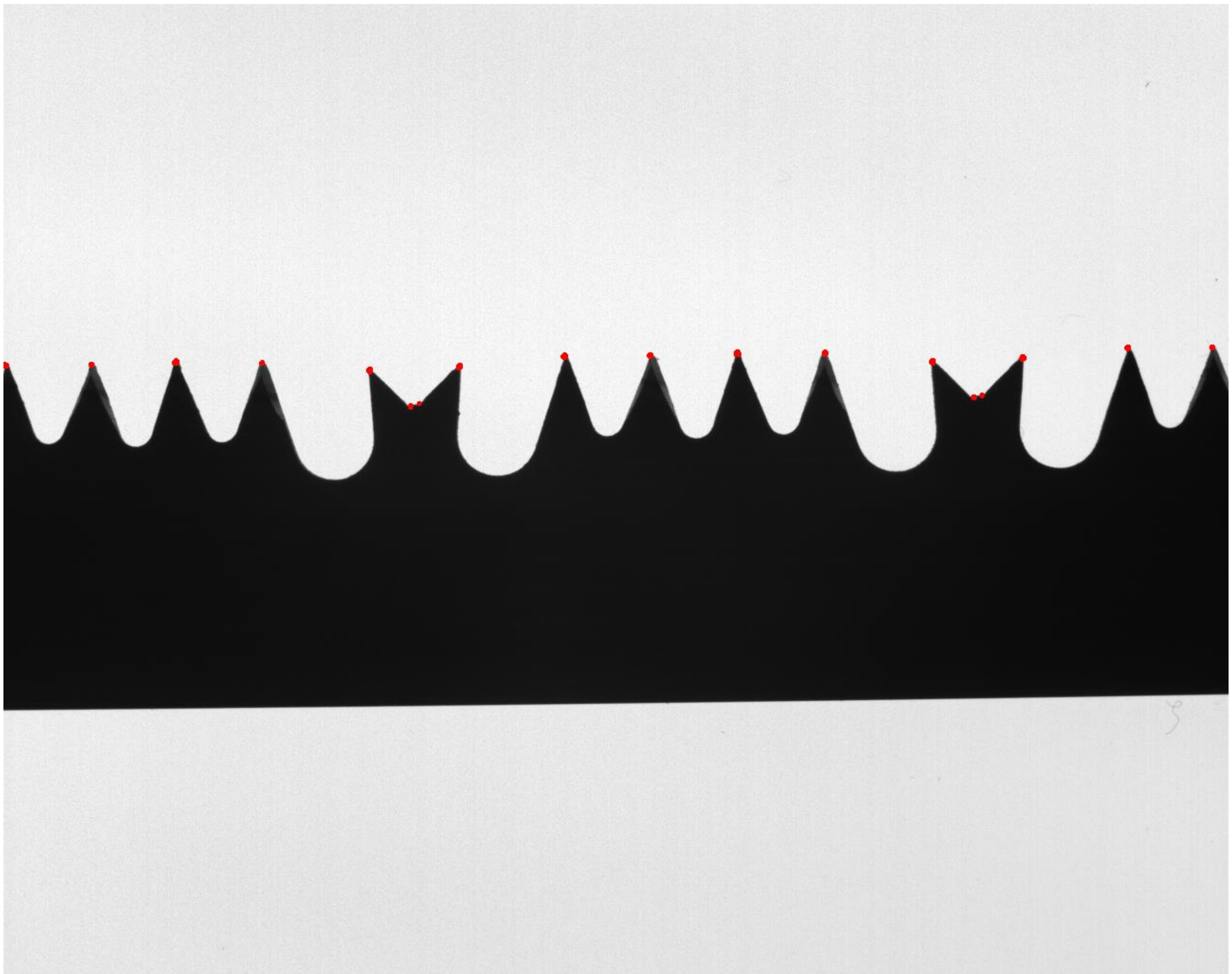
CHALLENGES

The task was of a geometric nature in the sense that I was required to detect triangles (a tooth is just a triangle with the base missing). Early on, it became apparent that the primary challenge would be to somehow reduce the image such that only the straight-line profiles were visible. Once that was accomplished, the next challenge was to approximate triangles by joining the lines in a geometrically sensible manner.

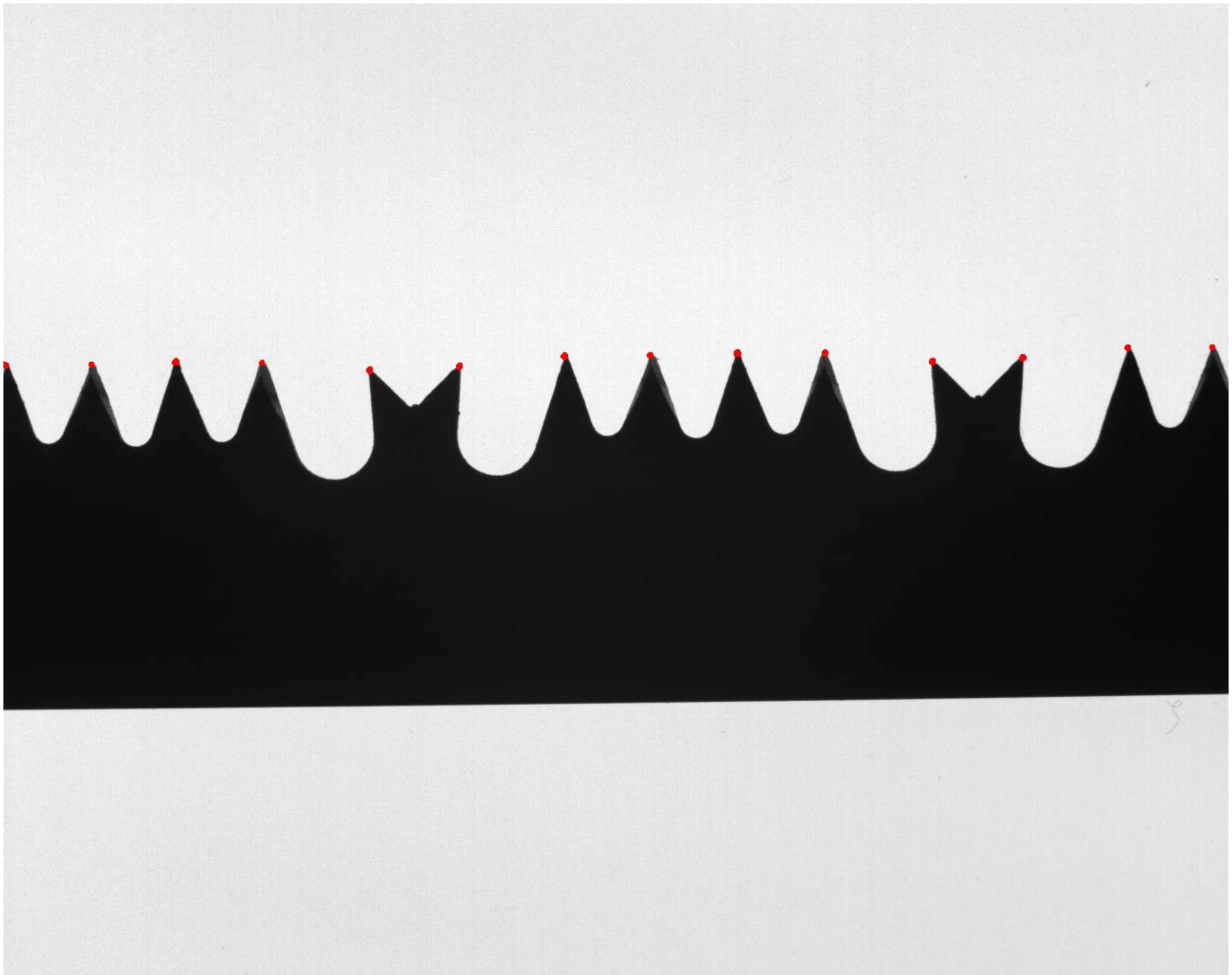
METHOD

The process of identifying the angles and verifying the rectilinear nature of each tooth profile is achieved through the following steps:

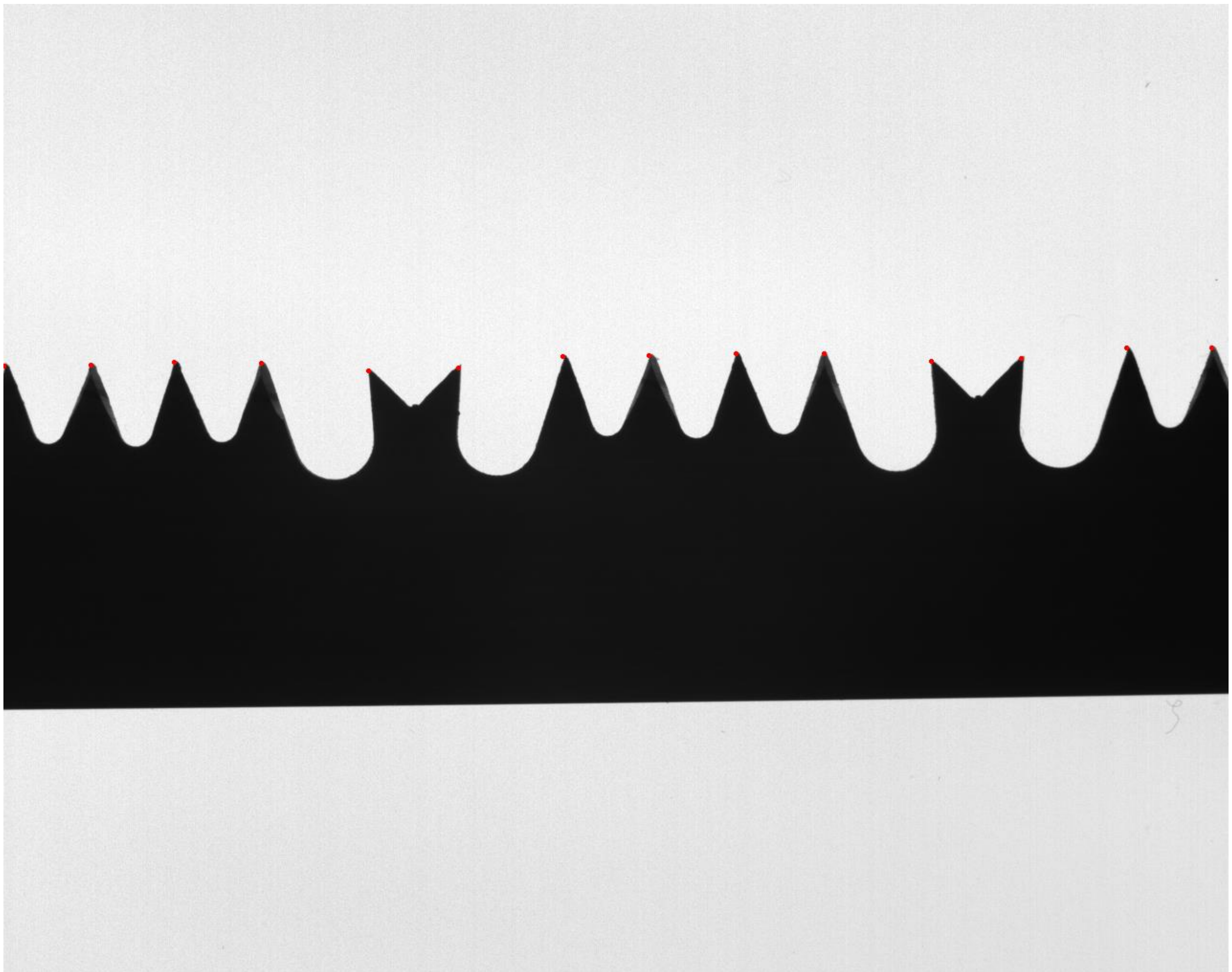
1. **Detect Corners:** Harris corner detector is applied on the gray scale image. As expected, most of the corners are obtained at the tip of each tooth with a few stray corners along the edge and curved troughs.



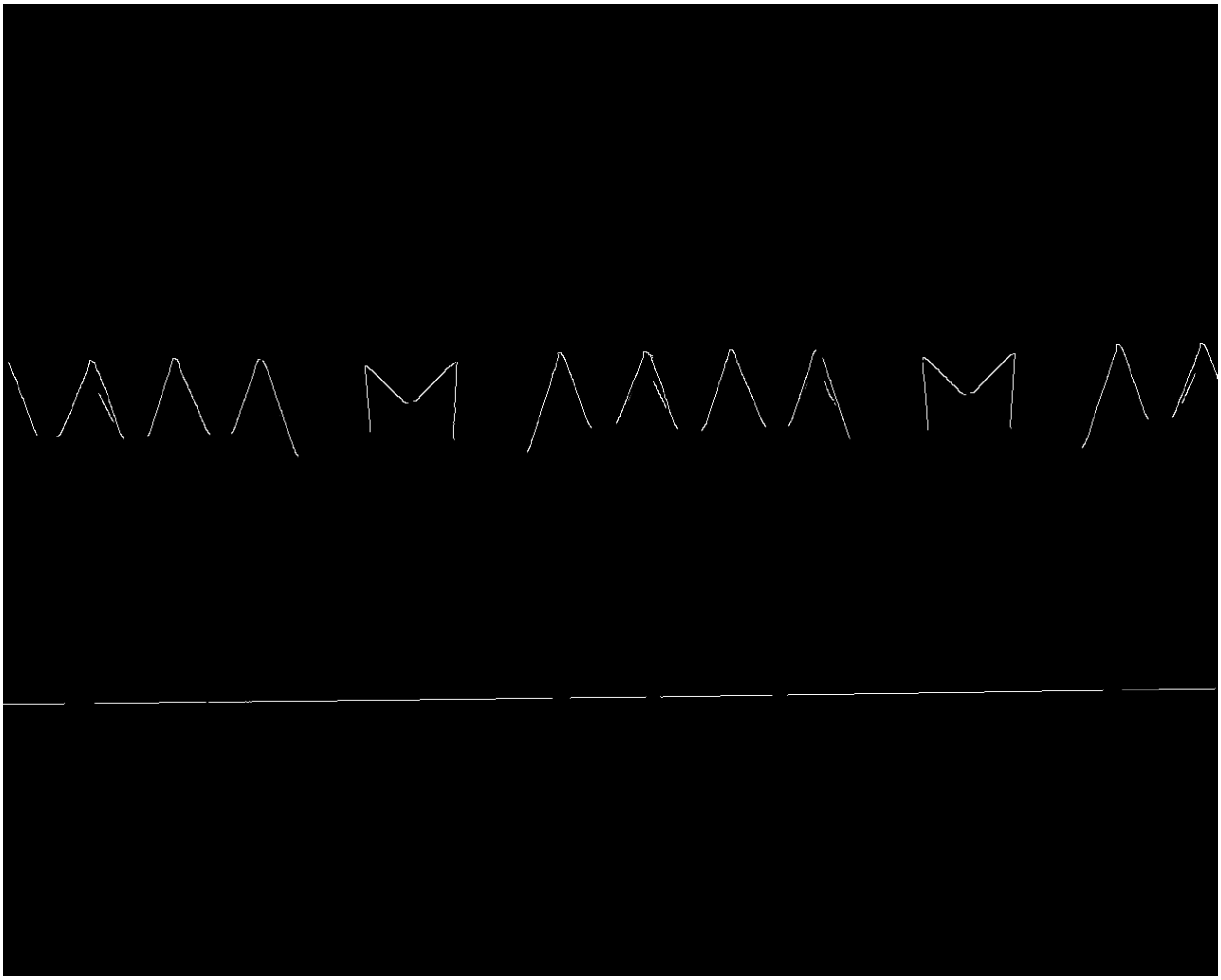
2. **Corner Filtering Stage 1:** To remove the unwanted corners, I simply compute the average y coordinate of all the corners. Then, corners that are too far to the bottom are removed. Adjusting the threshold for Harris will not have the same effect since some troughs in the image have small spikes jutting out. Hence, that will be treated as a legitimate corner no matter how much we adjust the threshold of the Harris corner detector. Therefore, it simply seems sensible to exploit the fact that such stray corners are too far to the bottom and must be discarded based on how far they are from the remaining corners.



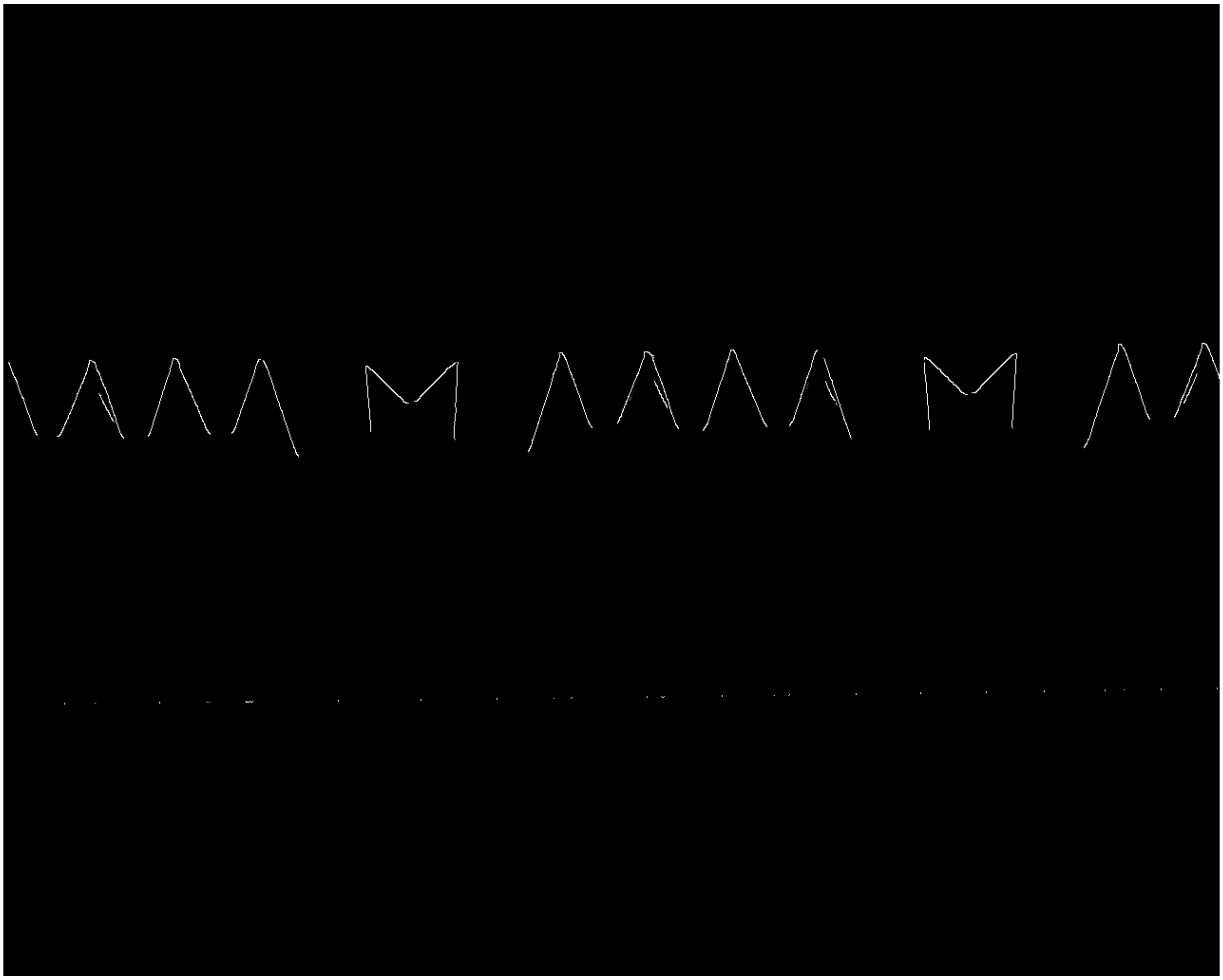
3. **Corner Filtering Stage 2:** Due to the imperfect nature of the peak and the algorithm of the Harris corner detector, we get several corners at each peak. Therefore, an extra filtering step is applied to make sure that every tip of a tooth has only one corner on it. For this, the corners are first arranged in ascending order of their x coordinate. Then, we sequentially iterate through the list of corners. Corners that are too close together are filtered such that only one corner is chosen out of them. In the interest of keeping everything simple (and because it doesn't matter which corner is chosen as all of them are so closely spaced together) the first corner out of the closely spaced corners is chosen.



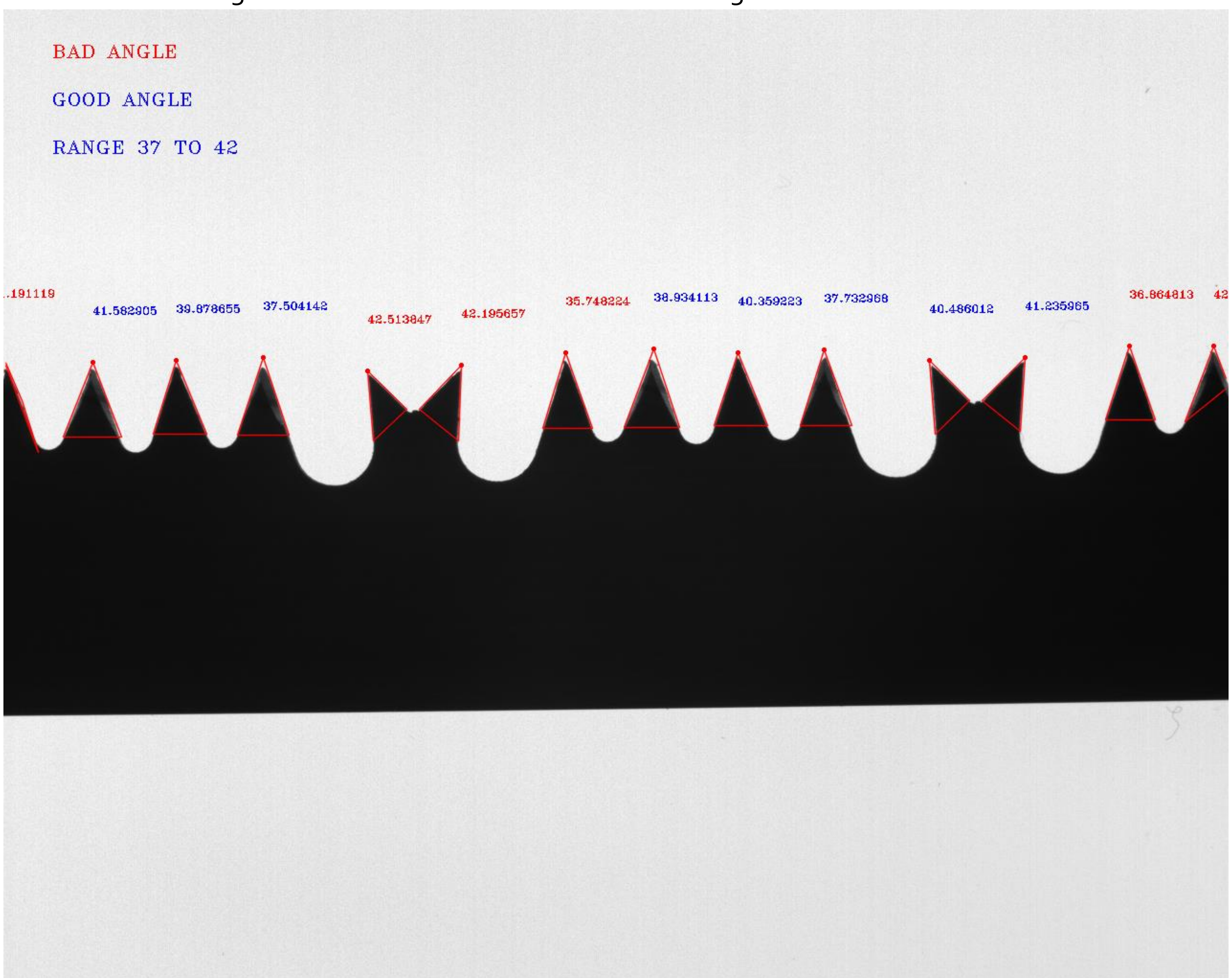
4. **Remove curved surfaces:** We now have a more appropriate list of corners. The next step is to extract the straight-line profile. For this purpose, we first extract edges from the original image using Canny's edge detector. Then, probabilistic Hough Line transform gives us the straight lines. We draw these straight lines on an empty single-channel image and take the bitwise AND of this image and the image with all the extracted edges from Canny's. The result is a new image with no curved surfaces on it.



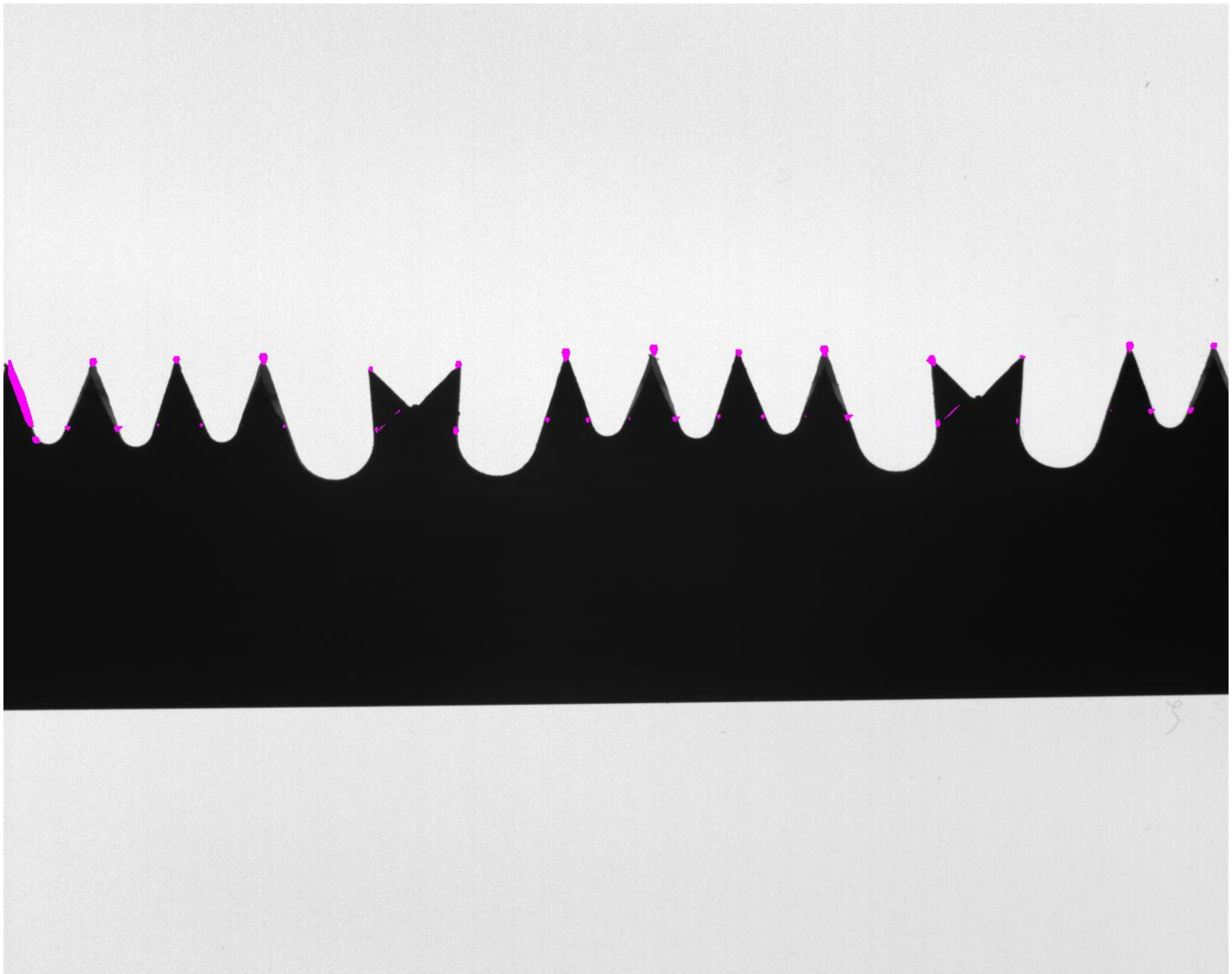
5. **Remove horizontal line:** An extra filtering step is applied to remove the horizontal lines at the bottom. We use a morphological operation to extract the horizontal lines and plot them on a separate single channel image. Then a bitwise XOR operation with the binary image from the previous step gives us a new image with no horizontal lines.



6. **Draw rectilinear profile:** We now have everything we need to approximate a rectilinear profile at each tooth. For a given detected corner, take its coordinate. Now, go to that exact coordinate in the binary image that only has the tooth lines drawn on it (you obtained this binary image at the end of the previous step). Search in the immediate vicinity of the coordinate for white pixels. The search area is a rectangle with a width and height that can be controlled by the user during runtime. Once the white pixels are obtained, you run the OpenCV function minEnclosingTriangle on them to draw the smallest possible triangle that contains all the points.
7. **Calculate angle:** For angle calculation you just get the coordinate of the top-most point of a given triangle. Then, we use the formula $\tan^{-1}\frac{m1-m2}{1+(m1*m2)}$ to get the angle at that point, where m1 and m2 are the gradients of the two lines that form the angle we want to measure.

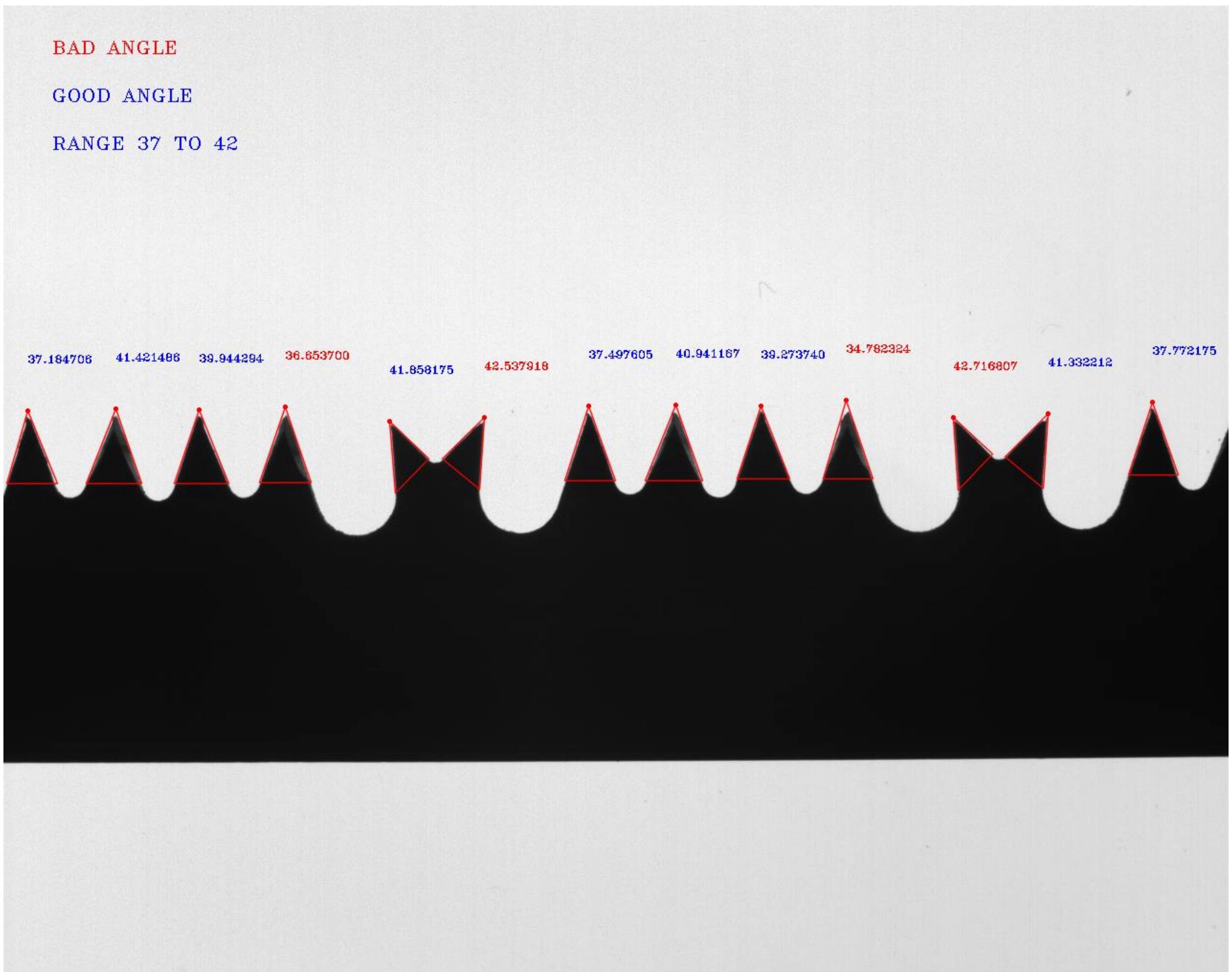


8. **<u>Indicate imperfections:</u>** An extra step involves pointing out the specific places with imperfections (points along the teeth that are not perfect). This simple process involves first drawing the triangles on the original binary image with just the straight lines of the tooth profile. Then, I convolve this new image with a 11 by 11 filter of 1s. The coordinates in the image where the convolution results in a value greater than a user-selected threshold (through numerous experimentations, I have decided to fix this threshold to 0.15), is highlighted on the original gray scale image with a pink dot.

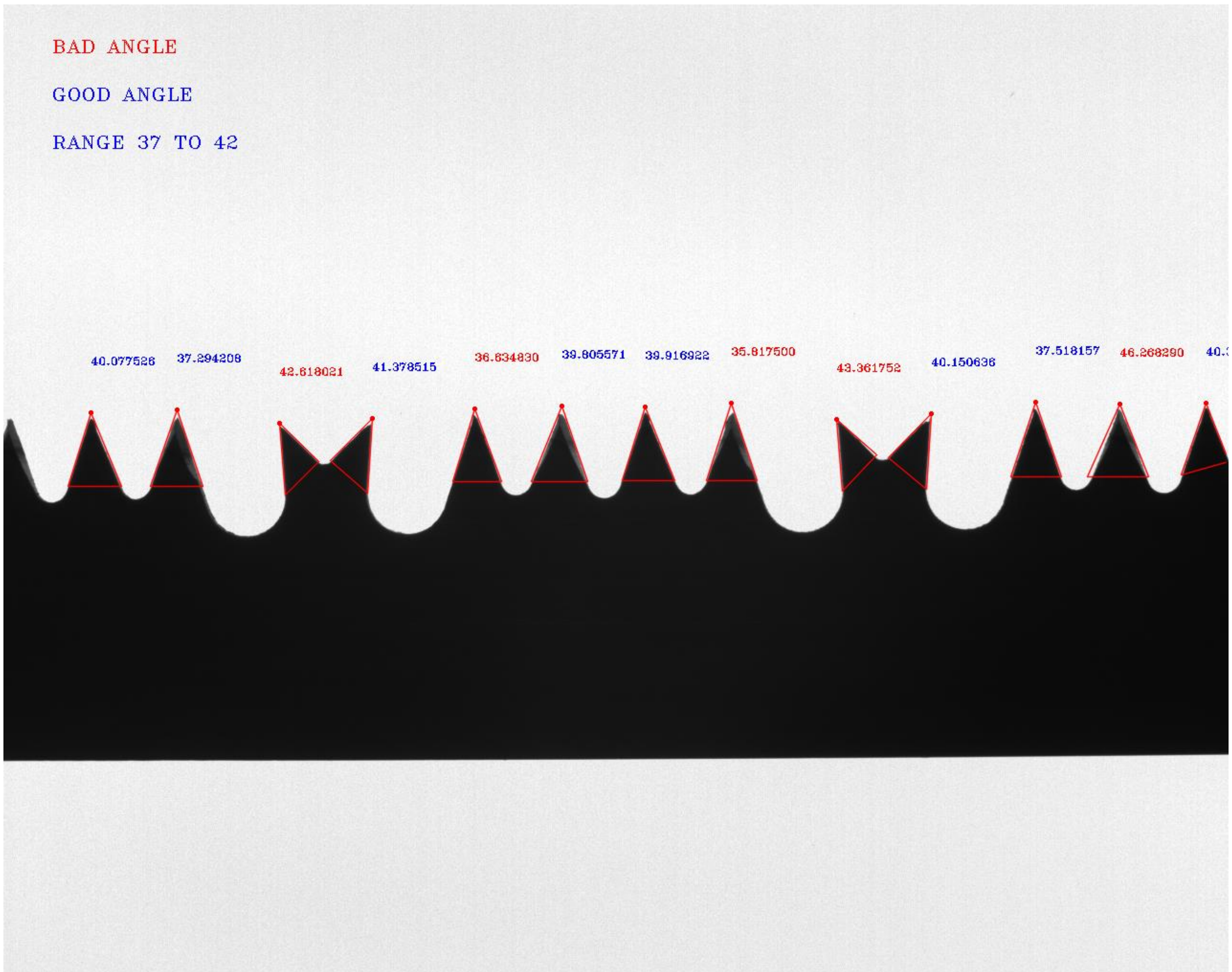


RESULTS

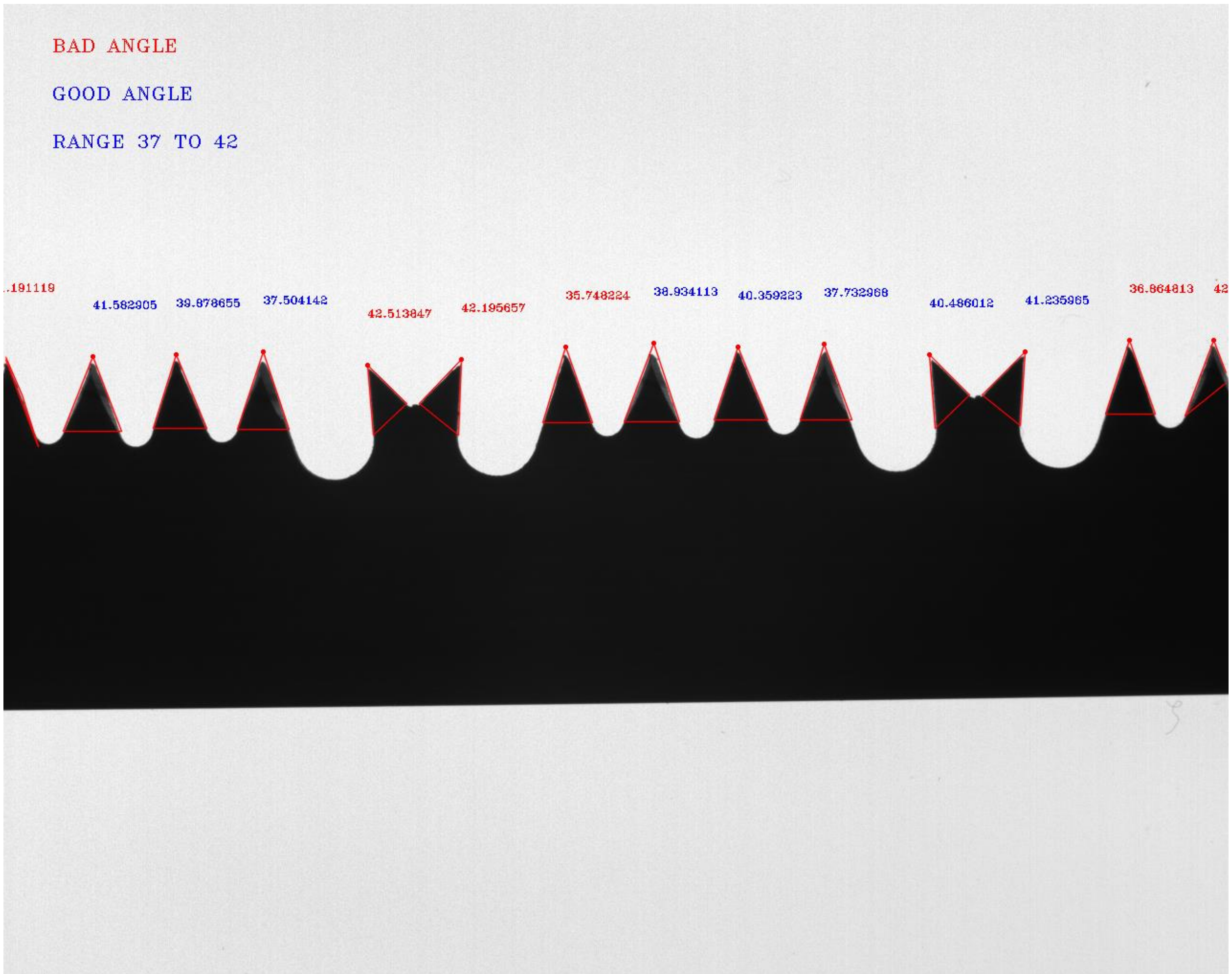
Saw_01.png



Saw_02.png

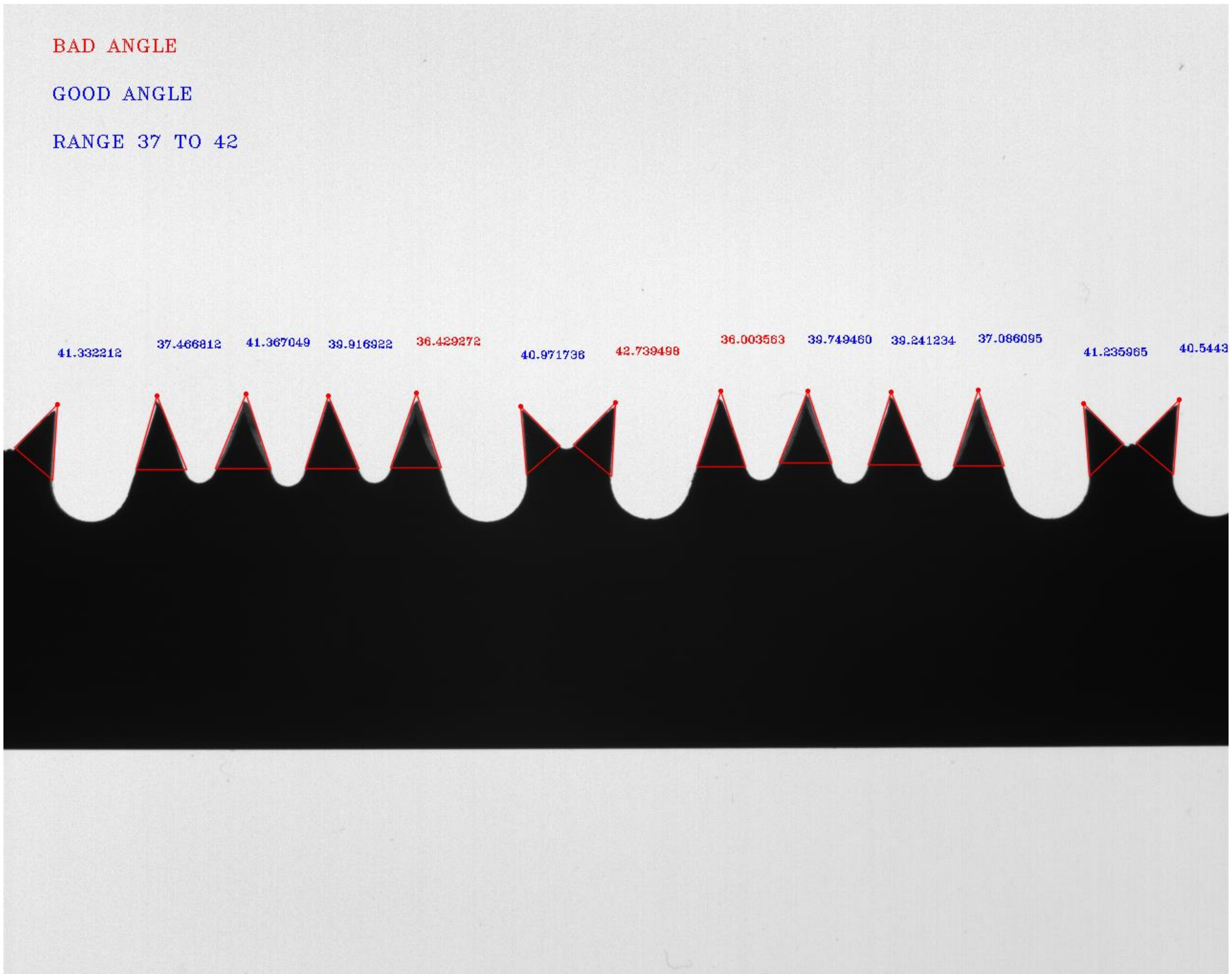

BAD ANGLE
GOOD ANGLE
RANGE 37 TO 42
40.077526
37.294208
42.618021
41.378515
36.634830
39.805571
39.916922
35.817500
43.361752
40.150636
37.518157
46.268290

Saw_03



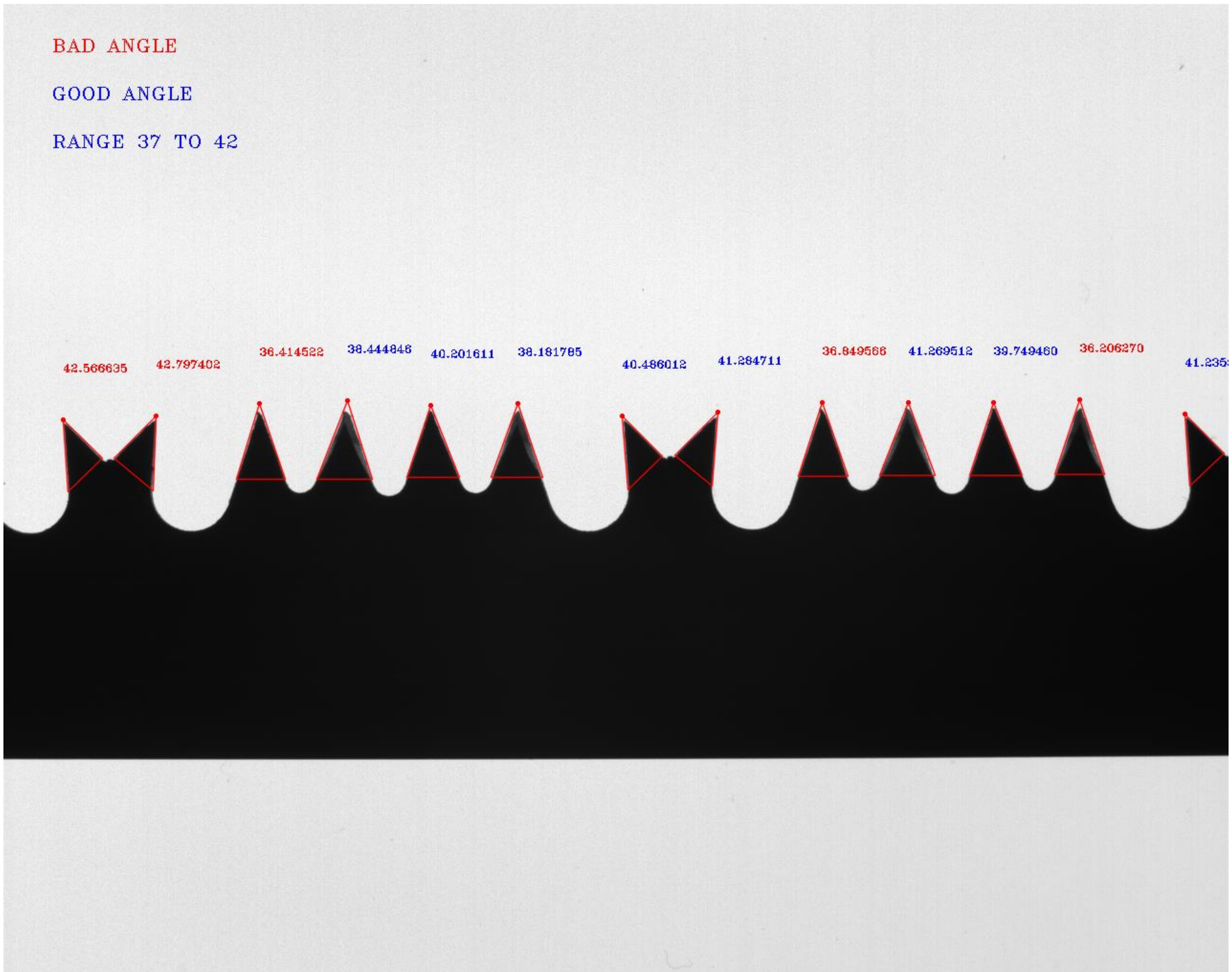
BAD ANGLE
GOOD ANGLE
RANGE 37 TO 42
.191119
41.582905
39.878655
37.504142
42.513847
42.195657
35.748224
38.934113
40.359223
37.732968
40.486012
41.235965
36.864813
42

Saw_04

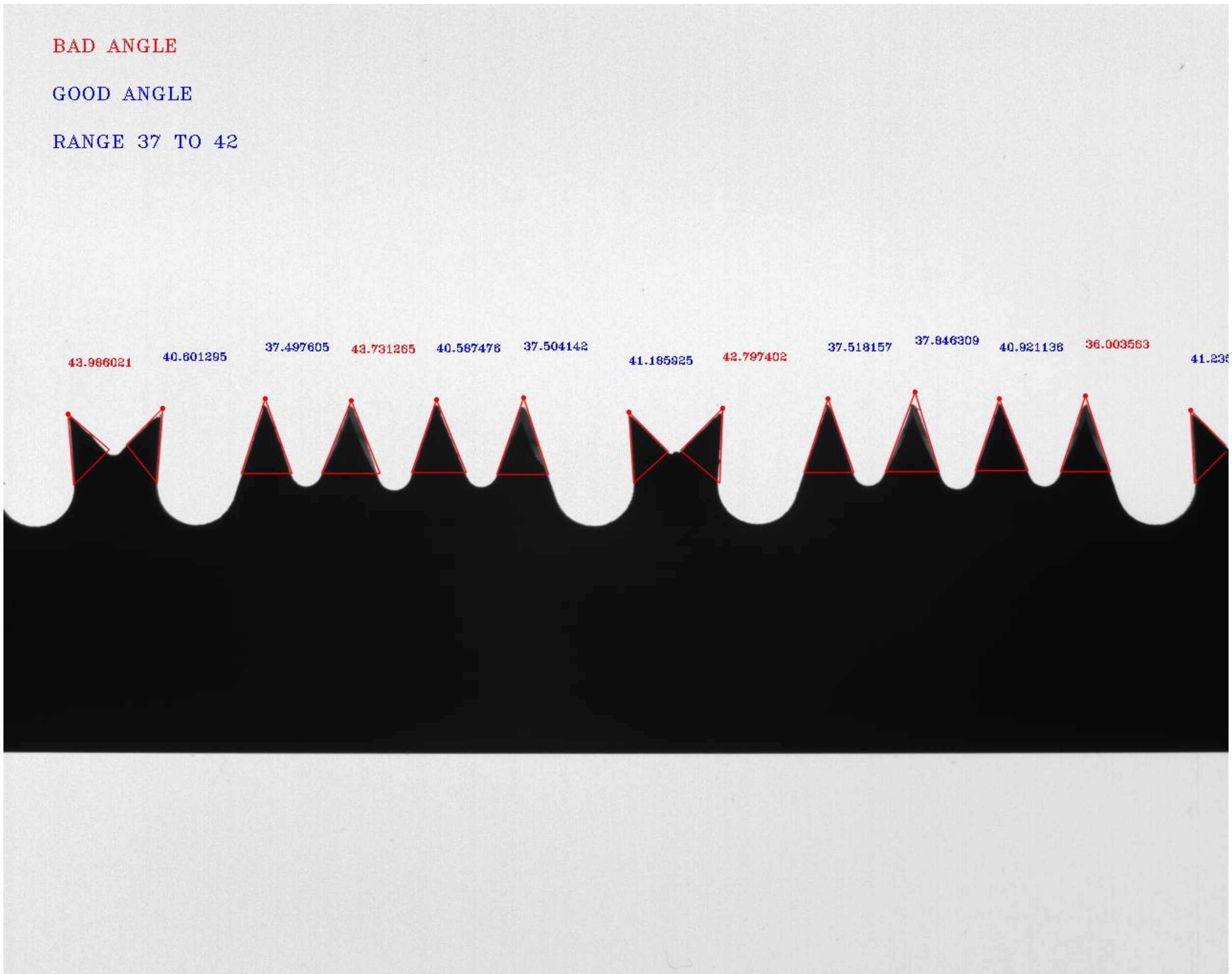


BAD ANGLE
GOOD ANGLE
RANGE 37 TO 42
41.332212
37.466812
41.367049
39.916922
36.429272
40.971736
42.739498
36.003563
39.749460
39.241234
37.086095
41.235965
40.5443

Saw_05

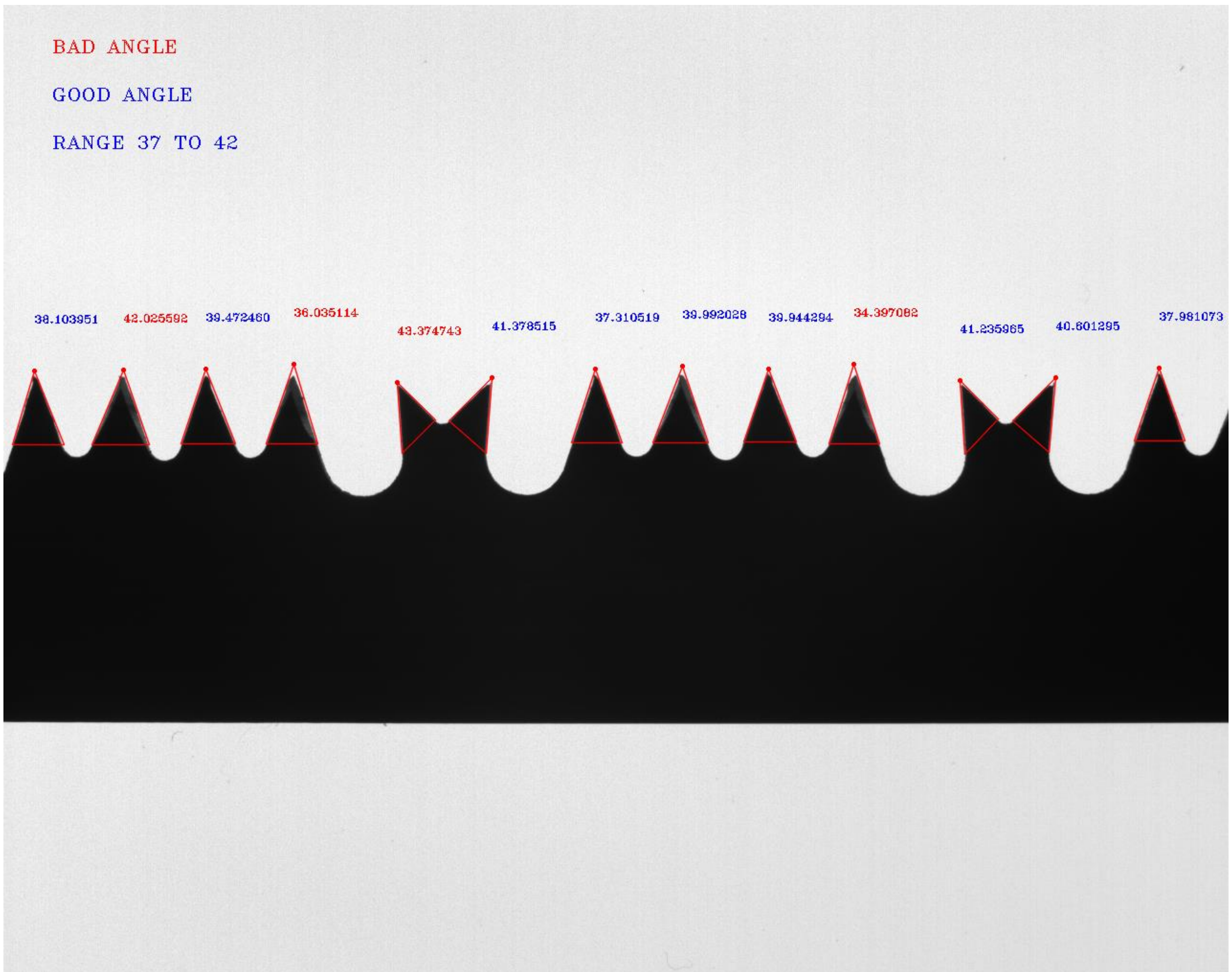

BAD ANGLE
GOOD ANGLE
RANGE 37 TO 42
42.566635
42.797402
36.414522
38.444846
40.201611
38.181785
40.486012
41.284711
36.849566
41.269512
39.749460
36.206270
41.235

Saw_o6



BAD ANGLE
GOOD ANGLE
RANGE 37 TO 42
43.986021
40.601295
37.497605
43.731265
40.587476
37.504142
41.185925
42.797402
37.518157
37.846309
40.921136
36.003563

Saw_07


BAD ANGLE
GOOD ANGLE
RANGE 37 TO 42
38.103951
42.025592
39.472460
36.035114
43.374743
41.378515
37.310519
39.992028
39.944294
34.397082
41.235965
40.601295
37.981073