Team Mensa Martiri Report

Automated Detection of Steel Bars Using Machine Vision and Deep Learning

Aliprandi Francesco, Degiorgi Nicola, De Castelli Fabrizio,
Loika Arseni, Sbarbaro Steven

Objectives and Approach

Objectives

Duferco was seeking an AI solution to streamline its steel production process. Specifically, they aimed to automate the detection of steel bar alignment against a stopper wall in rolling mill operations. This task, currently performed manually by operators, is crucial for ensuring accurate cutting and overall efficiency. By leveraging AI, we hope to reduce human error and increase production efficiency. The ideal solution should accurately classify images as 'aligned' or 'not aligned', ideally in real-time, handle variations in camera perspectives and lighting conditions, and be deployable on standard hardware.

Approach

The solution proposed starts with manual data annotation. For labeling, we chose to use `Label Studio` [1], classifying each edge of the beam as 'contact' if it touches the stop bar and 'non-contact' if it does not, as shown in Figure 1.

(a) Dark conditions, all aligned

(b) Light conditions, 1/8 not aligned

Figure 1: Labeling examples

For data cleaning we decided to first normalize the data, ensuring that all images are in same brightness ranges. Then we apply a Fourier transform to reduce high frequencies and slightly clean the image, thus improving visual quality and removing noise.

The core of the system is based on a pipeline consisting of two models placed in sequence. The task of the first model is to identify the upper corners of the steel bars while the second model has to classify those points into "aligned" or "not aligned" basing on their relative contact with the stop bar. By switching from the first to the second model, the system zooms in on the key points of the image, creating a patch of size 100×100 so that the second model can better focus on the key points.

The first model relies on a convolution-deconvolution architecture based on the pretrained TorchVision `ResNet` [2]. It takes cleaned images as input and returns the positions of key points of interest, specifically the edges of steel bars with and without contact with the stopper wall.

The second model is also a pretrained `ResNet` that we adapted through transfer learning to specialize in recognizing aligned and misaligned beams. We trained this model while reserving a holdout set for evaluation and identified the optimal training parameters using a grid search.

It is important to note that producing certain code was assisted by AI, primarily ChatGPT [3].

Finally, to make predictions on unseen images, the model takes the input, cleans it, and uses the first network to identify key points on the beam. For each identified key point, it creates a patch centered around that point, then classifies each patch. If at least one patch is classified as misaligned or there are no detected bars, the entire image is classified as misaligned otherwise we classify the image as aligned.

Challenges

There were many aspects of this challenge which made it incredibly complex to solve.

The first major challenge involved determining the optimal labeling strategy for our data. We wanted to label as precisely as possible while avoiding the need for additional information later that we hadn't marked. Initially, we wanted to simply label the images as 'aligned' or 'not aligned' and then use a CNN or another "black box" to predict the class. However, to both deepen our learning and explore alternative solutions, we decided on a more nuanced path: labeling the corners of the steel bars.

Later, we discovered that detecting the corners was much harder than we had anticipated. First, we tried classic Harris detection, but there was too much noise, making it difficult to isolate the needed points. We also attempted to find the top part of every steel bar with Canny edge detection, we then used Hough Transforms to recognize the line and its ending to predict the corners. This worked better but only recognized around 70/80% of the corners. In the end, we decided to find these points using a CNN, which worked well but still left room for improvement, especially since we trained on little data.

Another significant challenge involved the quality of the images themselves. The average quality of the steel bar images was suboptimal for a machine learning task, affected by camera quality, varying light/dark conditions, and equipment vibration that blurred some images. This required extra care in labeling and additional time to fine-tune our network. Dark conditions were particularly challenging, but we overcame this by changing the brightness value in HSV color space and by using a Fourier transform to enhance the bars' edges.

Lastly, the time constraint was a major challenge. Although we had many other ideas, we didn't have enough time to try them all, and some methods took longer to implement than expected. With more time, we would have added steps to the pipeline to address some edge cases that occur in this setting.

Results

Conv-Deconv Network Result

Finding the positions of the angles of the beams with this method has shown promising results. While the best trained model has been lost while trying to improve its performance, its effectiveness remains to be seen in the validation visualizations 2. The current best available model has worse performance, which quite significantly damages the classifier stage. The team is strongly believed that superior results can be achieved with this method by investing additional time in refining the training parameters, labeling additional data and correcting the network dimensions.

ResNet Classifier Result

This model is trained over 100×100 sized images. Best model is select over f-beta score but also precision, recall and accuracy are computed.

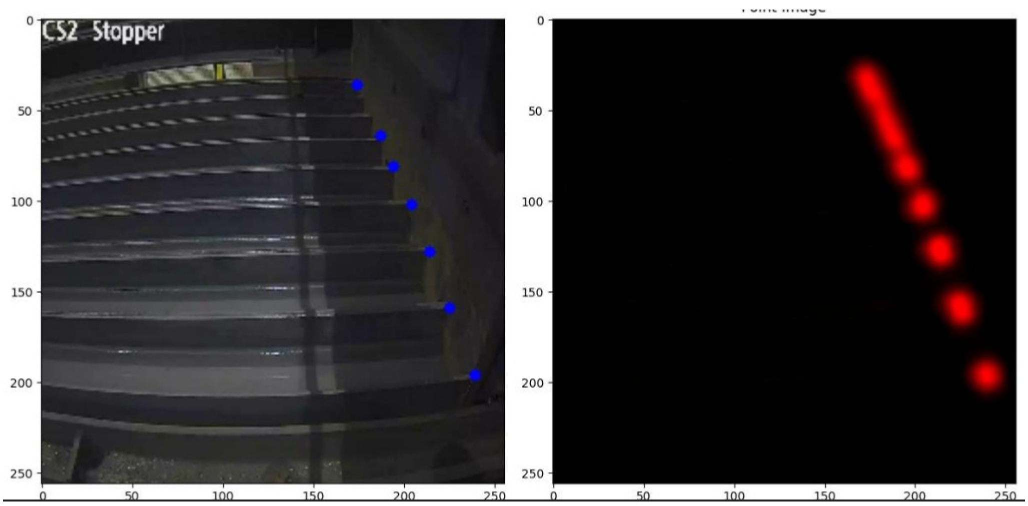

Figure 2: Corner detector result

Thanks to brightness normalization preprocessing and by focusing on areas of interest through zoomed-in sub-images, the model achieved improved classification results, with an F-beta score of 0.96, accuracy of 0.95, precision of 0.96, and recall of 0.97.

The confusion matrix in Figure 3 shows balanced performance across both classes, highlighting the model's accuracy without significant bias toward any category. This even distribution of errors suggests a well trained and robust model with consistent results across all classifications.

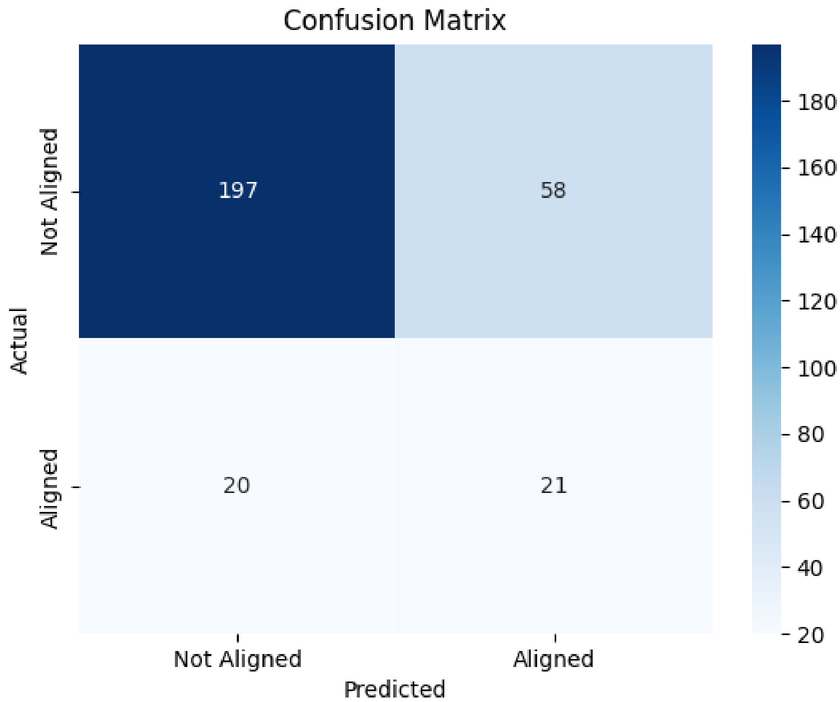

Figure 3: ResNet classifier confusion matrix

Complete Model Result

While the individual components of the model perform well separately, the overall efficiency of the integrated system is not as high as expected. The results show an F-beta score of 0.33, accuracy of 0.74, precision of 0.29, and recall of 0.28, indicating that the model's performance drops. However, the merging of these components has not yielded the expected improvement in performance, suggesting that further optimization is needed to enhance the synergy between the individual models.

References

[1] HumanSignal maintainers and contributors. *Label Studio: Open Source Data Labeling Platform*. `https://labelstud.io/`. 2020.

[2] TorchVision maintainers and contributors. *TorchVision: PyTorch's Computer Vision library*. `https://github.com/pytorch/vision`. 2016.

[3] OpenAI. *OpenAI ChatGPT*. `https://chatgpt.com/`. 2022.