# Methods for the localization of supporting slats of laser cutting machines in single images

Frederick Struckmeier[1,2], Philipp Blättner[2], and Fernando Puente León †[2]

[1] TRUMPF Werkzeugmaschinen GmbH + Co. KG
Johann-Maus-Str. 2, 71254 Ditzingen, Germany
[2] Karlsruhe Institute of Technology, Institute of Industrial Information Technology, Hertzstraße 16, 76187 Karlsruhe, Germany

**Abstract** The supporting slats of laser flatbed machines cause process reliability problems, such as tilted parts colliding with the cutting head. In order to mitigate these problems the position of the supporting points for a part to be cut must be known, before the machines numerical control program can be changed accordingly. Being able to detect the position of supporting slats accurately is necessary to do that. This work compares image processing methods to localize the supporting slats in single images. The best features are based on filters in the frequency domain and can have accuracies above 96 %.

**Keywords** Image processing, object detection, object localization, laser cutting, laser flatbed machine

## 1 Introduction

Laser flatbed cutting machines (LFMs) are an important part in the sheet metal production process, as they are able to efficiently cut contours of any form. In the LFM layout, the metal sheet is still, while the cutting head moves above it. Supporting slats are used to support the metal sheet during the cutting process. The slats are metal strips, that are a few millimeters wide and are basically a row

of isosceles triangles (see Fig. 1.1 left). The slats are attached to the pallet at certain positions, where the slat is pushed into a socket.

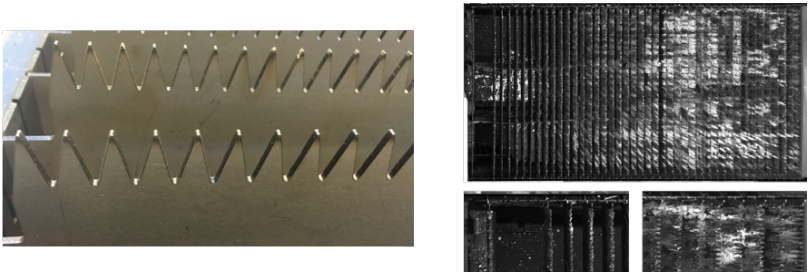

**Figure 1.1:** Left: Supporting slats of a LFM. Right: An example image of the empty pallet, with detailed sections of the left and right upper corners below.

Whilst being cost efficient and decently robust regarding the adverse conditions under the sheet being cut, the supporting slats cause some problems with the process reliability [1]. For example, a part may tilt after being cut free, depending on the position of supporting tips under that part and the gas pressure. A tilted part can cause collisions with the cutting head leading to downtime of the machine. Also, if cutting right above a tip, the slat might be damaged unnecessarily and the part can have lower quality due to visible marks [1].

In order to prevent these problems, adjustments to the numerical control program of the machine have been suggested, namely changes to the nesting layout [1] and the tool path [2]. However, these approaches assume that the position of the supporting tips relative to the raw metal sheet is known in advance. This is generally not the case with the LFMs in use today.

Reference [3] presented different methods to measure the supporting slats of LFMs. Whilst that work focused on laser triangulation, it pointed out a big advantage of detecting slats in single images: there is almost no auxiliary process time needed. As it is unclear which method is best suited to detect slats in an image, this work compares different methods.

The task is to find an estimator that is first able to identify the slat tips in the image and then translate this information to the slat socket positions of the pallet. A major difficulty is the possible varia-

tion in the appearance of the slats. They can be made from different materials, mostly mild steel, stainless steel and copper. Also, cutting causes wear and tear of the slats. Firstly, the drops of molten material exiting from the cutting kerf stick to the slats. A well-used slat therefore has multiple colours (see Fig. 1.1 right). A different cause of the variation in the images is the background. The machine has two pallets, so one might be above the other. The lower pallet can be seen through the upper pallet from the camera perspective. Also, in an industrial setting there are often scrap pieces of metal on the floor below the pallet, which can also be seen through the pallet.

In the next section we present the state of the art of object detection and localization in single images. In Section 3, the different features and classifiers of this work are explained in detail. The test set and the results of the methods are presented in Section 4, before Section 5 concludes the work.

## 2 State of the art

Whilst there is a comprehensive body of literature on object detection and localization in images, the problem of localizing supporting slats of LFMs in single images has never been studied before to the knowledge of the authors.

The definition of the terms object detection and object localization in different image processing works is not always the same. Sometimes these terms are used almost interchangeably, because predicting that an object is present in an image is usually based on features, whose appearance in the image can be restricted to certain locations of the image. For the rest of this work we will define object localization to include the detection and accurate estimation of the location of the searched object [4].

An established method for object localization is template matching. A known template is convolved with the image, resulting in the feature image. When detecting a single instance of an object in an image, classification is performed by selecting the highest peak [5].

Another approach to localization of an object in an image is parallel projection. In 2D images, the parallel projection is equivalent to summing the pixel values in a given direction or an integral over

that axis. It is used in some medical image processing works to localize a tool, e. g. a needle, in a 3D image obtained by ultrasound imaging [6].

Most other successful frameworks do not focus on object detection and localization in the sense of this paper. The SIFT algorithm [7] for example can find different known objects at different scales and rotations. However, in our case there is only one object of the same size and problems arise because of high variance in lighting, background and wear conditions.

The most recent and for many use cases very successful method is applying convolutional neural nets for image processing. Because there are only about 200 images available for training and validating such a framework, this approach is not further pursued.

## 3 Methods

### 3.1 Image Acquisition and Perspective Transformation

The camera taking the images is mounted on top of the LFM overlooking the pallet outside of the machine body (see Fig. 3.1). The perspective requires a camera with a wide-angle lens.

**Figure 3.1:** The position of the camera on the machine body.

The image perspective is transformed, so that it displays the scene from a birds-eye view. The resulting image (see Fig. 1.1 right) has a size of 1600 by 3100 pixels. Note that the perspective transformation leads to the slats close to the machine body being seen from above, whereas the slats on the other side of the pallet are seen at an angle.

### 3.2 Features

Intuitively, slats are vertical lines in the images. The triangular shape of the tips leads to many edges and corners along the slat. They can also be seen as a texture. Hence, the features selected for further study are different edge and corner detectors, Laws' energy measures and a hand-crafted model of slats in the spatial frequency domain. In order to establish a baseline, the unmodified images are also used as an input to the two classifiers introduced in Section 3.3.

#### Edge and Corner Detectors

An edge in image processing is simply put a large change in brightness along a line in the image. The change in brightness can be detected by analyzing the first or second derivative of pixel values. Hence, two approaches are tested, namely the gradient-of-Gaussian filter and the Difference-of-Gaussians (DoG) filter.

The first is an approximation based on the gradient of the image. It can be shown that a smoothing of the image with a Gaussian low-pass filter and a differentiation of the image is equal to a convolution of the image with the derivative of a Gaussian low-pass filter [5]. Discrete sampling of the Gaussian low-pass filter will result in a discrete formulation of the gradient-of-Gaussian filter.f

The Laplacian-of-Gaussian filter is an approximation of the second derivative of the smoothed image [8], which can be used for edge detection. The Laplacian filter is the simplest approximation of the second derivative obtained by a convolution. However, it is rather sensitive to noise, which is why the image is smoothed with a Gaussian low-pass filter. A discrete implementation is approximated by the DoG filter, which is based on the difference of two Gaussian functions with different standard deviations [5].

For corner detection, the Harris corner detector is used [9]. The idea for this detector is to take a small window of the image and test how much change happens to the values if the window is shifted by a small distance in all directions. In an area with no edges or corners, the change in pixel values will be low. If an edge is present, the change will be low in direction of the edge. At a corner however, there is significant change in all directions, when the window

is moved. A more formal and detailed description can be found in [5,9].

### Laws' Energy Measures

Another approach is to interpret the slats as textures. One of the most used texture processing algorithms are Laws' energy measures [10]. They consist of a set of quadratic matrices of variable size. The matrices of size $5 \times 5$ were chosen, as they were shown to be a good trade-off between information content and computational speed [10]. The matrices are defined as the result of all combinations of outer products of four vectors, representing the detection of levels ($\mathbf{l_5}$), spots ($\mathbf{s_5}$), ripples ($\mathbf{r_5}$) and edges ($\mathbf{e_5}$):

$$\mathbf{l_5} = (1,4,6,4,1)^T, \qquad \mathbf{s_5} = (-1,0,2,0,-1)^T,$$
$$\mathbf{r_5} = (1,-4,6,-4,1)^T, \qquad \mathbf{e_5} = (-1,-2,0,2,1)^T.$$

The matrix resulting from the multiplication of $\mathbf{l_5}$ with itself is disregarded, because it calculates a weighted average. The final set consists of 15 matrices. The features are defined as the energy of the convolution of the filter matrices with the image. The resulting image $g(u,v)$ is convolved with a Gaussian low-pass filter of size $5 \times 5$, referred to as $f_1$ to decrease high-frequency noise. To combine the $N = 15$ feature images, the average is calculated.

### Features in the spatial frequency domain

As the slats can only be placed at certain distances and the tips have a given distance between them, one would expect certain spatial frequencies in the Fourier space to have peaks.

The tips and sinks of a slat form vertical lines in the image and have a certain distance. This results in symmetric horizontal lines with varying intensity in the frequency domain. The distance $d_1$ of the first of the symmetric lines to the axis can be calculated from the vertical distance of two supporting tips $d_{\text{tip}}$, the height of the image $h$ and the size of a pixel $\Delta x$ [5]: $d_1 = \frac{1}{d_{\text{tip}}} \Delta x h$.

These expected horizontal lines can clearly be seen in the frequency domain (see Fig. 3.2 left) and do occur at the predicted

distances. Since there are more lines of higher frequencies, three band-pass filters (BP1 to BP3) are defined to extract the signal in the frequency space. The range of values for $f_x$ is limited by a boundary $b_x$, as the energy of the signal decreases with higher frequencies.

$$\mathrm{BP1}(f_x, f_y) = \begin{cases} 1, & \text{if } 30 < |f_x| < b_x \text{ and } |f_y| < 5 \\ 0, & \text{otherwise} \end{cases}$$

$$\mathrm{BP2}(f_x, f_y) = \begin{cases} 1, & \text{if } |f_x| < b_x \text{ and} \\ & d_1 - 5 < |f_y| < d_1 + 5 \\ 0, & \text{otherwise} \end{cases}$$

$$\mathrm{BP3}(f_x, f_y) = \begin{cases} 1, & \text{if } |f_x| < b_x \text{ and} \\ & 2 * d_1 - 5 < |f_y| < 2 * d_1 + 5 \\ 0, & \text{otherwise} \end{cases}$$

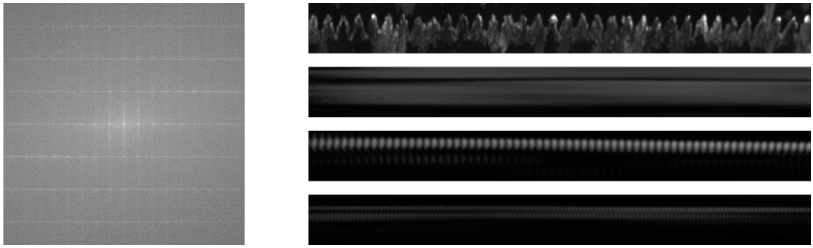

**Figure 3.2:** Left: An example section of the Fourier magnitude spectrum with the origin in the center. For better visibility of the characteristic features, the leakage effect was reduced by multiplying the original image with a Hann window. Right: An example section of a slat. From top to bottom: pixel values, BP1, BP2 and BP3.

After applying one of the filters, the image is transformed with the inverse DFT, resulting in an image containing only those pixels of the filtered frequency.

### 3.3 Classifiers

Two classifiers were used to extract the slat positions from the feature images. The parallel projection is intuitive in this case, as the slats are vertical lines in the images. Template matching is a widespread method for object detection and localization.

### Parallel Projection

One simple and intuitive way of detecting objects that span across the whole y-direction is parallel projection. Summing the pixel values of the feature image along the y-axis results in a discrete signal that should have peaks at the object locations.

The wavelet transform based pattern matching method presented in [11] is used to extract these peaks. First, a convolution of the signal with discretized Ricker wavelets of 20 different widths is calculated. The response of this convolution is stored in a matrix with each row describing a different wavelet. Next, the peaks in this matrix are analyzed within a row and across adjacent rows. These peaks are referred to as ridge lines. They can be filtered for, giving the position of the peaks.

The advantage of this method is that peaks of different sizes can be detected easily and that the shape information of each peak is not lost, leading to higher information efficiency.

### Template Matching

A widely used method for detecting objects in a signal is template matching, which has a two-step approach [12]. Firstly, a template is generated either by example or hand-crafting. Secondly, the occurrence of the template in a given image is evaluated using a similarity measure, e. g. the sum of absolute differences or the normalized cross correlation (NCC) [12]. In this case the NCC was used, since it is almost independent to changes in brightness or contrast of the image [13].

The simplest template to use for this use-case is the image of a new slat. Since the perspective changes from the left to the right of the image, a slat from the middle of the image is taken.

For frequency space features, the template is taken from the filtered and inverse transformed image. A new slat from the middle of the palette is used and will be referred to as “filtered slat template”. A problem occurring quite often with this feature were double peaks, where both the sinks and tips of a slat cause a peak in the NCC. One way to compensate this is to crop the left part of the template since the right side of a correct peak is mostly dark, but to the left there

might be wrongly detected sinks. This template will be referred to as "asymmetric filtered slat template".

The other template used is a simple binary mask, that detects a bright area in the middle of two black areas. The masks must have a suitable width in pixels $w_s$, given the width of a slat in the image:

$$BM(u,v) = \begin{cases} 1, & \text{if } \frac{1}{3}w_s < v < \frac{2}{3}w_s \\ 0, & \text{if } 0 \leq v \leq \frac{1}{3}w_s \text{ and } \frac{2}{3}w_s \leq v < w_s \end{cases}$$

#### Transformation to slat positions

There is a given number of possible slat positions on the pallet, as slats can only be inserted in their socket. Therefore, a transformation is needed to map from the 3100 image columns to 93 slat socket positions. Since each image might have a slightly different angle or calibration, a general transformation from x-positions to slat positions is not possible. In order to calculate this transformation one needs to extract the position of the sheet stop $x_{\text{stop}}$ from the images. The minimum distance between two slats $d$ is roughly the same as the distance between the slat stop and the first slat. The sheet stop position was extracted using template matching on the original image analogous to the description above in a small area in the upper left region of the image. The template was taken from a single image for each of the two test sites.

The position $n \in \{1, 2, \ldots, 92, 93\}$ of a slat detected at a certain x-position $x_n$ is calculated as:

$$n = \text{round}\left(\frac{x_n - x_{\text{stop}}}{d}\right).$$

## 4 Evaluation

### 4.1 Results

A test set of 215 images acquired from TRUMPF TruLaser 3000/5000 machines was used to evaluate the methods. 27 images were taken in a TRUMPF test setting and 188 images at a test customer site. The pallets of all machines are almost identical and have 93 slat sockets. Both the slats and the sheet stop were labeled manually.

**Table 1:** Accuracy results of different features with a parallel projection based classifier.

| Feature | True pos. rate | True neg. rate | Accuracy |
|---|---|---|---|
| Pixel values | 0.86 | 0.486 | 0.667 |
| BP1 | 0.855 | 0.412 | 0.625 |
| BP2 | 0.916 | 0.655 | 0.781 |
| BP3 | 0.897 | 0.805 | 0.85 |
| Laws Energy Measures | 0.903 | 0.453 | 0.67 |
| Harris Corner Detector | 0.605 | 0.396 | 0.497 |
| Difference-of-Gaussian | 0.88 | 0.581 | 0.725 |
| Gradient-of-Gaussian | 0.807 | 0.336 | 0.562 |

The evaluation of the methods is based on a binary vector with 93 elements. For every classification result, the accuracy is calculated as:

$$\text{accuracy} = \frac{\text{no. of true pos.} + \text{no. of true neg.}}{\text{no. of classifications}}.$$

As the data set is quite balanced between empty and full slat positions, this metric is applicable.

The true positive rate, true negative rate and accuracy can be seen in Tables 1 and 2.

**Table 2:** Accuracy of different features with a template matching based classifier.

| Feature | Template | True pos. rate | True neg. rate | Accuracy |
|---|---|---|---|---|
| Pixel values | New slat | 0.204 | 0.642 | 0.434 |
| BP2 | Filtered slat | 0.948 | 0.948 | 0.948 |
| BP2 | Asymmetric filtered slat | **0.966** | **0.958** | **0.961** |
| BP1 | BM | 0.764 | 0.715 | 0.738 |
| BP2 | BM | 0.921 | 0.857 | 0.887 |
| BP3 | BM | 0.93 | 0.87 | 0.898 |

### 4.2 Discussion

Generally, methods using one of the frequency line features performed better than any other method tested. The best method with

a different feature is the DoG with 72.5 % accuracy. Methods using the frequency line features score around 90 % quite often.

Using operations on the pixel values themselves had a better performance with parallel projection than template matching. This seems to indicate that the variation in lighting and wear condition is high. Otherwise it would be easier to find a good template and thus feature images are needed.

Edge and corner detectors as well as Laws' energy measures yield accuracies between 56 % and 72 %. Most likely they are sometimes confused by structures in the background as well as slag formations on the slats. Also, they use no information about tip distances, a definite disadvantage compared to the frequency line features.

Mistakes are mostly made in that part of the pallet, where the slats are seen at an angle, regardless of which method is used. One might expect that this error pattern can be reduced if the template for template matching classifier is taken from this area of the image. A test showed slightly worse results though, most likely because the variations in the appearance of the slats have a greater effect if more of the side of the slat is visible.

## 5 Conclusion

In this work, different methods for localizing slats of LFMs have been tested on a data set from different machines. Whilst edge and corner detectors, texture measures and operations on the pixel values showed accuracies between 45 % and 70 %, features based on the spatial frequency were best to extract the information and lead to an accuracy of up to 96.1 %.

For this study, neural nets were disregarded, as there are rather few images. With more images from different sites it might become feasible to train neural networks for this task in the hope that they will be better in suppressing relevant noise.

Another direction for future work is the fusion of information across images. In this study, every image was treated by itself. However, since the pallets of a single machine do not change much over time, there might be valuable information that can be extracted from the image sequence.

## References


1. F. Struckmeier and F. Puente León, "Nesting in the sheet metal industry: dealing with constraints of flatbed laser-cutting machines," *Procedia Manufacturing*, vol. 85, pp. 149–158, 2019.
2. R. Dewil, P. Vansteenwegen, and D. Cattrysse, "A review of cutting path algorithms for laser cutters," *The International Journal of Advanced Manufacturing Technology*, vol. 87, no. 5-8, pp. 1865–1884, 2016.
3. F. Struckmeier, J. Zhao, and F. Puente León, "Measuring the supporting slats of laser cutting machines using laser triangulation," *The International Journal of Advanced Manufacturing Technology*, vol. 108, no. 11, pp. 3819–3833, 2020.
4. A. Andreopoulos and J. K. Tsotsos, "50 years of object recognition: Directions forward," *Computer Vision and Image Understanding*, vol. 117, no. 8, pp. 827–891, 2013.
5. J. Beyerer, F. Puente León, and C. Frese, *Machine Vision: Automated Visual Inspection: Theory, Practice and Applications*. Berlin, Heidelberg: Springer, 2016.
6. M. Barva, M. Uhercik, J.-M. Mari, J. Kybic, J.-R. Duhamel, H. Liebgott, V. Hlavác, and C. Cachard, "Parallel integral projection transform for straight electrode localization in 3-D ultrasound images," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 55, no. 7, pp. 1559–1569, 2008.
7. D. G. Lowe, "Object recognition from local scale-invariant features." in *IEEE International Conference on Computer Vision*, vol. 99, no. 2, 1999, pp. 1150–1157.
8. D. Marr and E. Hildreth, "Theory of edge detection," *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 207, no. 1167, pp. 187–217, 1980.
9. C. G. Harris, M. Stephens *et al.*, "A combined corner and edge detector." in *Alvey Vision Conference*, vol. 15, no. 50. Citeseer, 1988, pp. 10–5244.
10. K. I. Laws, "Texture energy measures," in *Proceedings: Image Understanding Workshop*, 1979, pp. 47–51.
11. P. Du, W. A. Kibbe, and S. M. Lin, "Improved peak detection in mass spectrum by incorporating continuous wavelet transform-based pattern matching," *Bioinformatics*, vol. 22, no. 17, pp. 2059–2065, 2006.
12. G. Cheng and J. Han, "A survey on object detection in optical remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 117, pp. 11–28, 2016.

13. K. Briechle and U. D. Hanebeck, "Template matching using fast normalized cross correlation," in *Optical Pattern Recognition XII*, vol. 4387. International Society for Optics and Photonics, 2001, pp. 95–102.