

PRIP-TR-146

May 9, 2019

# Interactive segmentation and contour extraction for automatic phenotyping of Lipizzaner horses<sup>1</sup>

*Daniel Pucher, Walter G. Kropatsch*

## Abstract

In this paper we introduce a new custom software that was developed to support researchers in their study of exterior movement characteristics of the Lipizzaner horses. It uses an interactive image segmentation algorithm to segment a single horse in a given image from the background. The interactive algorithms offer a good tradeoff between accuracy and user interaction and allows faster segmentation than manual tools. Based on the resulting foreground mask of the horse, the whole contour or a segment of it can be exported for further analysis.

---

<sup>1</sup>Supported by FFG BRIDGE 1 "Automatisation der Phänotypisierungen von Exterieur-und Bewegungsmerkmalen beim Lipizzaner"

# 1 Introduction

Researchers at the University of Veterinary Medicine in Vienna are interested in the automated phenotyping of exterior movement characteristics of the Lipizzaner horse. In order to study the exterior characteristics, a special setup is used to capture an image of a single horse in a normalized pose. The pose is predefined in order to make results of different horses comparable.

To make the processing of this specific image data easy and fast, we developed a custom software that uses an interactive segmentation algorithm to segmented the horse from the background. Based on the segmentation, the contour of the horse can be extracted and exported to enable the analysis of the pose. A robust point on the contour is automatically selected as a start point for every horse.

## 1.1 Camera Setup

To capture an image of a horse in a pose, a video camera is placed on a tripod in front of a static background that should preferably be homogeneous and have a good contrast to the horse. A single horse is moved in front of the camera, parallel to the camera image plane and is guided by a trainer until it stands in the desired pose. In order to make the capture of this pose easier, a video of the horse is taken during this process and a single frame with the desired pose can be selected in the software afterwards.

## 1.2 Structure of the report

The remainder of this report is structured as follows. Section 2 gives a brief overview of the developed software. Section 3 presents challenges, a short literature review of state of the art segmentation algorithms and gives a more in depth view on the image processing approaches used in our software. Results are shown in Section 4. Finally, Section 5 presents the conclusion and future work.

## 2 Software

The software is written in Python with Qt<sup>1</sup> as a frontend and uses OpenCV<sup>2</sup> for image processing. The main window of the software (see Figure 1) allows the user to browse for a video folder, open a video and select an image for image processing by either opening an existing image or choosing one from an opened video.

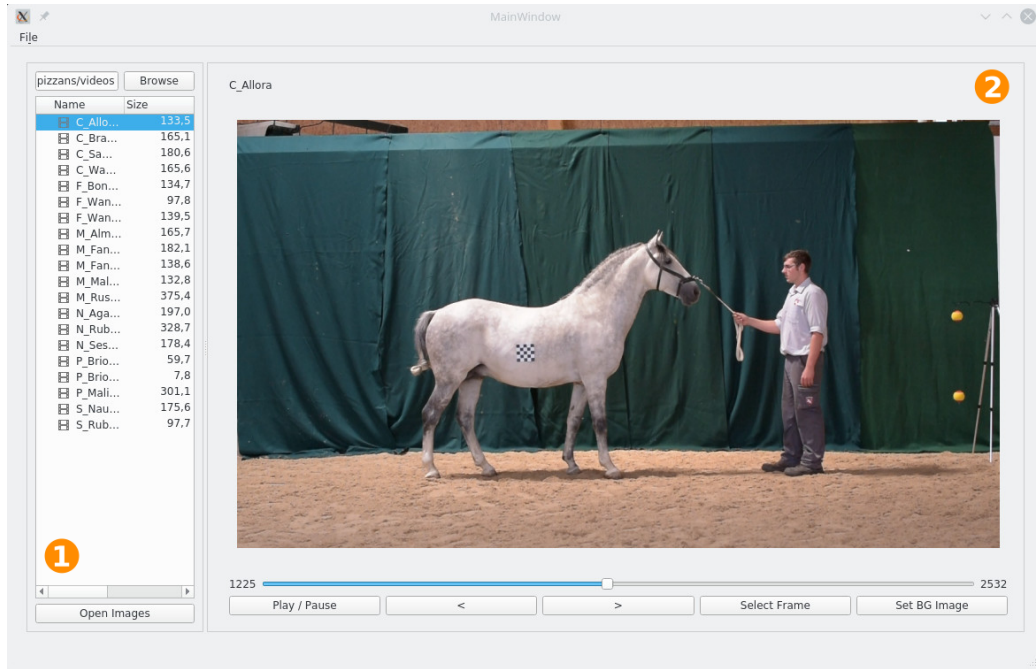


Figure 1: Main window. 1) Browse video folder and open video images, 2) Video controls

Opening an already existing image or selecting a frame from the video will open the image processing window (see Figure 2). This window offers controls to segment the horse from the background and export contour points from the resulting foreground mask. For the segmentation two algorithms are available.

The first one is a simple static background subtraction algorithm [13]. It uses the first frame of a video where no foreground objects are present as

---

<sup>1</sup><https://www.qt.io/>

<sup>2</sup><https://opencv.org/>

a static background model and computes the difference between this background image and the current image. In order to create a binary image out of the difference image, a threshold can be selected by the user.

The second algorithm is an interactive segmentation method that gives the user control over the segmentation process and is described in detail in Section 3.3.

Once the horse is segmented from the background, the contour of the horse can be computed and exported. For the export either the whole contour or a segment can be chosen. A segment is defined by the user with two end points.

For the whole contour, a start point has to be chosen. The start point can be detected automatically at a robust position or one of the contour points can be selected by the user as a start point. For the automatic detection, see Section 3.4. Finally, contour points are exported in a csv file with x and y coordinates in counterclockwise order. The number of contour points for the export can be selected by the user.

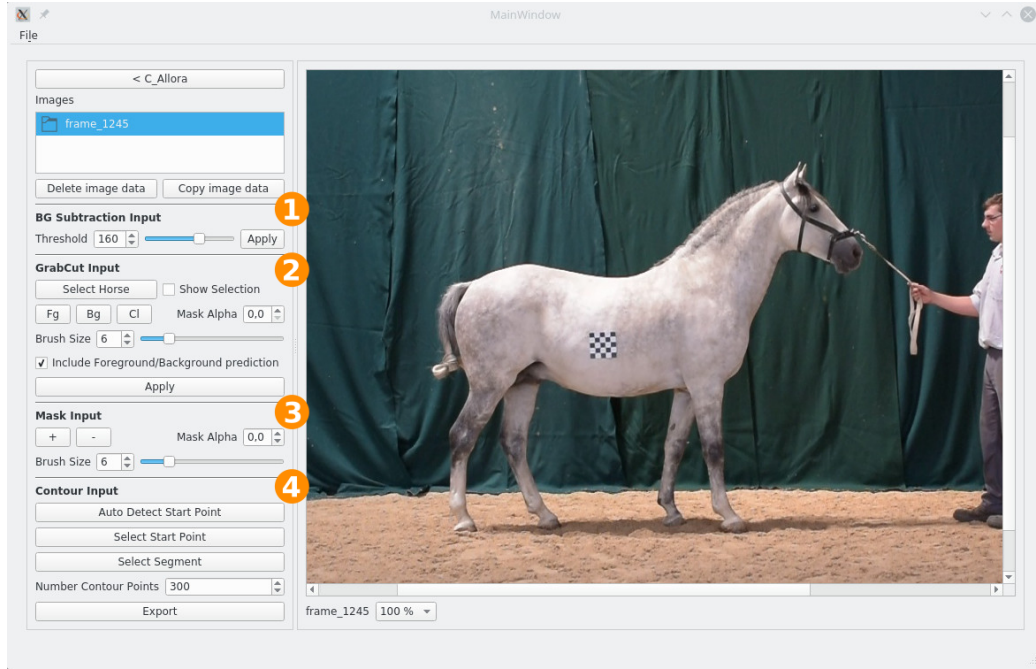


Figure 2: Image porcessing window. 1) BG Subtraction input, 2) GrabCut input, 3) Mask input, 4) Contour input

## 3 Image processing

### 3.1 Challenges

Depending on the background, the segmentation process can be challenging. Figure 3 shows three example images where the leg of a horse and the background exhibit similar color and texture information. Expert knowledge about the horse anatomy is needed in order to ensure an accurate segmentation.

Furthermore, since researchers are studying exterior movement characteristics based on the contour which is derived from the foreground region of a horse, the segmentation needs to be very accurate.



Figure 3: Example images of hard to segment areas close to the ankle and hoof of the horse.

### 3.2 State of the Art

In image segmentation a digital image is divided into regions that share significant properties, like foreground and background. Extensive research has been devoted to this area resulting in a number of different approaches. Heckel et al. [6] define the following groups of segmentation methods based on their degree of automation:

1. Fully automatic methods, where no user intervention is required.
2. Semiautomatic methods, where user inputs are given to the algorithm, e.g. indicating the object of interest.
3. Interactive methods, where the user enhances computer-generated segmentation results.
4. Manual tools, where the user has to paint the object of interest by hand.

Out of these methods, fully automatic ones are the goal since they give reproducible results and minimize the work that has to be done by the user [6].

For fully automatic or semiautomatic segmentation, approaches based on deep learning such as SegNet [1] or DeepLab [3] for natural image segmentation or SegAn [15] in the medical image domain are among the top performing state-of-the-art methods.

For interactive segmentation, GrabCut [12] is a well-known approach that has been extended over the years. Lempitsky et al. [8] extended it by incorporating a topological prior and Cheng et al. [4] improved it by the use of a fast fully connected Conditional-Random Field. Furthermore DeepCut [11] extends the GrabCut approach with a trained neural network classifier. Another interactive segmentation method based on deep learning has been introduced by Wang et al. [14] where a neural network model is made more adaptive with the use of image-specific fine tuning.

Based on the challenges mentioned in Section 3.1 the goal is to choose an interactive segmentation approach that is easy to implement and allows experts to steer the segmentation approach in an easy and fast manner. Furthermore we decided against a deep learning approach, since a usecase specific training dataset is not available. Based on these demands, we chose GrabCut [12], since it allows the user to steer the segmentation process in critical areas while also providing a good initial segmentation, minimizing the need for user interactions. Additionally the algorithm is already part of OpenCV and therefore easy to integrate.

### 3.3 GrabCut

GrabCut is an interactive segmentation algorithm proposed by Rother et al. [12] that extends an approach based on graph cuts developed by Boykov and Jolly [2]. It operates on color instead of grey scale images and uses a more powerful, iterative version of the optimisation. Furthermore, it significantly simplifies the user interactions necessary to achieve a certain quality of the segmentation by allowing incomplete labelling of foreground and background regions in the image. This allows the user to only specify the background region, which is determined as a pixel strip on the outside of a rectangle selected by the user. Although this initial, incomplete labelling can be sufficient for an entire automatic segmentation, the algorithm also supports further user input in the form of brushing pixels to be either foreground or background.

In order to increase the accuracy of the initial segmentation and reduce the user interaction, we start the algorithm with labels of automatically detected parts of the foreground and background. In our example camera setup, the background exhibits a bi-modal distribution. This results from the different colors and textures of the floor and wall. Therefore the whole image exhibits a tri-modal distribution between the floor, wall and horse. Since this is true for many possible setups, we make use of this information and use the extension of Otsu’s method [7] [9] to multi-level thresholding to separate the horse from the background.

The resulting binary image is processed with morphological dilating and eroding in order to create two masks, one that is completely inside the horse, and another that is completely outside the horse. The two resulting masks are used as an initial input for the GrabCut algorithm, eliminating the need for the user to mark a rectangle in the image as well as improving the accuracy of the initial segmentation. See Section 4 for results.

### 3.4 Contour Start Point

For the exported contours of different horses to be comparable, a robust start point on the contour has to be chosen. We chose the corner point between the front leg and the torso of the horse, since the contour exhibits a steep angle at this position. In order to automatically detect this point, the contour image is divided into regions and a Harris corner detector [5] is applied to one of the divided regions (see Figure 4) to find the most prominent corner.

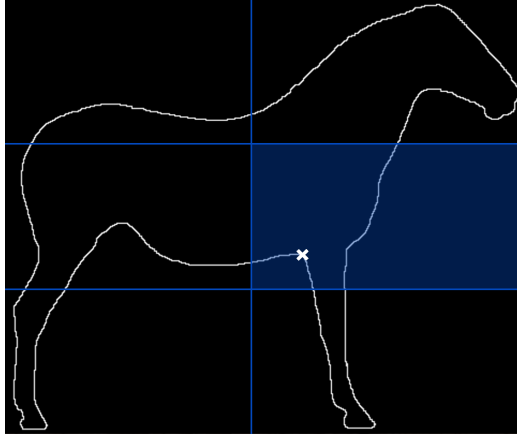


Figure 4: Start point detection with Harris corner detection in the blue region of the contour image. Start point marked with a cross.

## 4 Results

We tested our segmentation approaches on a dataset of eight images of different horses standing in the desired pose taken from different videos.

First we measured the time to complete a foreground segmentation and compared the interactive segmentation with GrabCut to a manual segmentation approach, where the whole foreground mask has to be drawn by the user. On average, the foreground segmentation with the manual approach took around three minutes. Using GrabCut, the time could be reduced to around one minute. This gives GrabCut a speed improvement of around 65% with fewer user interactions, compared to a manual segmentation approach.

Furthermore, we tested the segmentation results by comparing the initial results of GrabCut with two other methods. The first is a static background subtraction algorithm (BG-Sub.) with optimal threshold selected by the user. The second method is a two-stage Otsu (Otsu) where the threshold is selected automatically and the end result is enhanced by morphological opening and closing in order to remove noise.

In order to compare the different algorithms, the distance between a given segmentation and the ground truth is measured with a distance metric. We chose to use the binary Segmentation Edit Distance ( $SED_2$ ) [10]. This distance is based on the cost of edit operations that are needed to transform one given segmentation into another. In our case it measures the distance



Table 1: Results of the BG-sub., Otsu and GrabCut algorithms measured with the  $SED_2$  distance on the dataset (best scores are bold per score for every line).

|              | $SED_2$ |        |               |
|--------------|---------|--------|---------------|
| Horse        | BG-Sub. | Otsu   | GrabCut       |
| c_allora     | 40.346  | 47.067 | <b>11.305</b> |
| c_bradamanta | 23.569  | 32.362 | <b>5.307</b>  |
| c_samira     | 28.076  | 48.716 | <b>13.564</b> |
| c_wanda      | 26.884  | 35.337 | <b>12.524</b> |
| m_fantasca_b | 24.195  | 43.232 | <b>10.092</b> |
| m_malina     | 21.687  | 41.753 | <b>7.287</b>  |
| m_rustica    | 31.297  | 29.613 | <b>11.136</b> |
| n_aga        | 25.710  | 51.153 | <b>9.1</b>    |

between a given initial segmentation and the desired ground truth and is particularly suitable since our software allows the user to perform edit operations in order to improve the initial segmentation. The smaller the distance from a given segmentation to the ground truth, the less user interactions and time are needed in order to improve the initial segmentation. Table 1 shows the results comparing the three approaches.

## 5 Conclusion

In this paper we introduced a new custom software for interactive image segmentation in the context of studying exterior characteristics of Lipizzaner horses. For this task, images of horses standing in a certain pose are captured, since researchers are interested in the contour of the horse for further analysis. In order to extract the contour, the horse needs to be separated from the background first.

The implemented interactive segmentation algorithm allows the user to segment foreground objects by automatically generating an initial segmentation which can be improved by the user by the means of brush strokes, to mark foreground and background regions, and editing of the foreground mask on a pixel level. This makes the task of foreground segmentation both

easier and faster than manual segmentation, where the whole foreground mask is painted by hand. Comparisons to simple fully and semi automatic methods show better initial segmentation results, that can be further improved by user inputs.

Based on the foreground mask, contour points can be exported as a list of points with a robust start point that can be detected automatically. This makes contours of different horses comparable.

## 5.1 Future Work

In order to capture a horse in a certain pose, it is guided by a trainer in front of a camera and a video of its movement is captured. From such a video sequence, currently a single frame is taken for segmentation, contour extraction and further analysis. In the future we want to extend this approach to video segmentation which has two benefits: First, it would allow the extraction of contours of a whole video sequence allowing the researchers more in depth analysis of the horses in movement. Second, making use of all video frames could improve the segmentation of individual frames if taken into account, since parts that are occluded or flawed in a single frame can be improved by the same parts visible in other frames.

## References

- [1] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.
- [2] Y. Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *Proceedings eighth IEEE international conference on computer vision. ICCV 2001*, volume 1, pages 105–112. IEEE, 2001.
- [3] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018.

- [4] M.-M. Cheng, V. A. Prisacariu, S. Zheng, P. H. Torr, and C. Rother. Denscut: Densely connected crfs for realtime grabcut. In *Computer Graphics Forum*, volume 34, pages 193–201. Wiley Online Library, 2015.
- [5] C. G. Harris, M. Stephens, et al. A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Citeseer, 1988.
- [6] F. Heckel, J. H. Moltz, H. Meine, B. Geisler, A. Kießling, M. DAnastasi, D. P. dos Santos, A. J. Theruvath, and H. K. Hahn. On the evaluation of segmentation editing tools. *Journal of Medical Imaging*, 1(3):034005–034005, 2014.
- [7] D.-Y. Huang, T.-W. Lin, and W.-C. Hu. Automatic multilevel thresholding based on two-stage otsus method with cluster determination by valley estimation. *International journal of innovative computing, information and control*, 7(10):5631–5644, 2011.
- [8] V. Lempitsky, P. Kohli, C. Rother, and T. Sharp. Image segmentation with a bounding box prior. In *2009 Ieee 12th International Conference on Computer Vision (Iccv)*, pages 277–284. IEEE, 2009.
- [9] N. Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66, 1979.
- [10] D. Pucher and W. G. Kropatsch. Segmentation edit distance. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 1175–1180. IEEE, 2018.
- [11] M. Rajchl, M. C. Lee, O. Oktay, K. Kamnitsas, J. Passerat-Palmbach, W. Bai, M. Damodaram, M. A. Rutherford, J. V. Hajnal, B. Kainz, et al. Deepcut: Object segmentation from bounding box annotations using convolutional neural networks. *IEEE transactions on medical imaging*, 36(2):674–683, 2017.
- [12] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM transactions on graphics (TOG)*, volume 23, pages 309–314. ACM, 2004.
- [13] A. Sobral and A. Vacavant. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Computer Vision and Image Understanding*, 122:4–21, 2014.

- [14] G. Wang, W. Li, M. A. Zuluaga, R. Pratt, P. A. Patel, M. Aertsen, T. Doel, A. L. David, J. Deprest, S. Ourselin, et al. Interactive medical image segmentation using deep learning with image-specific fine tuning. *IEEE transactions on medical imaging*, 37(7):1562–1573, 2018.
- [15] Y. Xue, T. Xu, H. Zhang, L. R. Long, and X. Huang. Segan: Adversarial network with multi-scale  $l_1$  loss for medical image segmentation. *Neuroinformatics*, 16(3-4):383–392, 2018.