Rat Paw Tracking for Detailed Motion Analysis

Tobias Palmér Kalle Åström
Centre of Mathematical Sciences
Lund University
{tobiasp,kalle}@maths.lth.se

Olof Enqvist
Dep. of Signals and Systems
Chalmers University of Technology
olof.enqvist@chalmers.se

Nela Ivíca Per Petersson
Dep. of Experimental Medical Sciences, NRC
Lund University
{nela.ivica,per.petersson}@med.lu.se

***Abstract*—This work is part of a research project studying the learning of fine motor skills in rats. A system for tracking a rat paw using a system of high-speed cameras is presented along with preliminary analysis of the correlation between paw movement and neural data. The tracking method is generative, modeling the rat paw as a set of linked ellipsoids. To find the most probable paw pose a number of hypothetical parameter values are explored. For each set of values the paw model is projected in the different cameras and compared to the actual measured images. The pose that is most consistent with the intensity and edge information in the different views is chosen. By efficient utilization of integral images, this evaluation can be performed very quickly, creating a tractable and completely automatic method with good performance.**

I. Introduction

A. Background

The central nervous system fundamentally deals with the control of actions. Consequently behavioral studies have often been a natural starting point for investigations aimed at understanding its functions. The goal of this work is to study neuronal activity of the central nervous system during learning of new behaviours. As shown in [1], we have developed a system for analysis of motor behaviour of rats during skilled reaching experiments in order to learn about the changes occurring in the central nervous system.

There are several ways to improve the system from [1]. Firstly, evaluation of the quality function has been improved from quadratic time to linear time, with respect to the length of the shortest side of the videos. This has made available more advanced methods for quality function optimization, i.e. both tracking quality and computational efficiency has been improved. Secondly, the depth of each projected quadric is used in order to detect occlusions from the experimental setup. This makes the similarity measures used for quality evaluation more appropriate. As this is work-in-progress, there are not yet any behavioural recordings available using the latest experimental setup, thus there are no results presented using the new setup.

B. The behavioural task

In the behavioral experiment the rat was placed in the reaching apparatus from where it could obtain 45 mg food pellet rewards, positioned in an indentation on the reward shelf, via controlled reaches through the aperture of the wall. At the start of the experiment, the rats are slow and inefficient in this task, but after approximately two weeks of almost daily training, they have fully learned the reaching behaviour. This is interesting from a neural point of view as we then can study what happens in the brain as the rats learn this new behaviour.

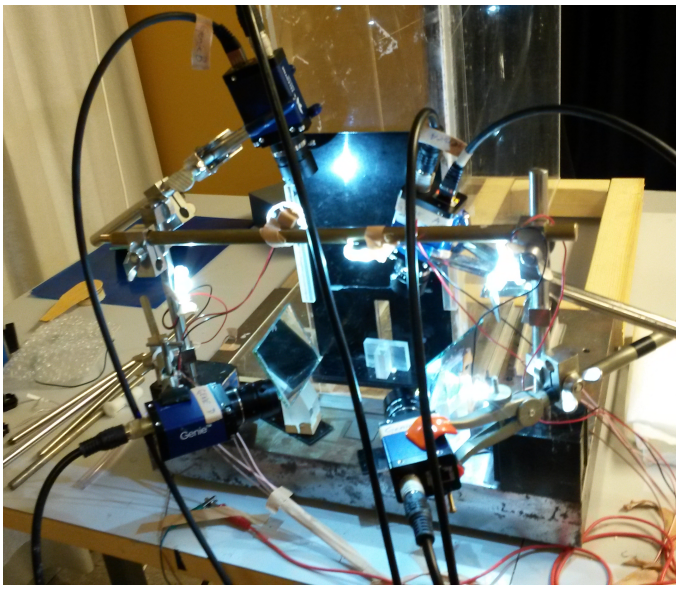


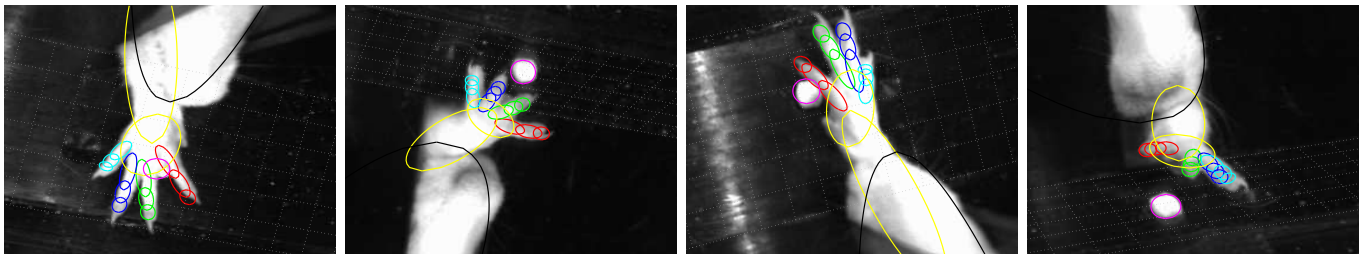
Fig. 1: The experimental setup with four cameras, lights and a calibration object.

C. The videos

An improvement from the setup in [1], is the use of four front-view cameras, as seen in Figure 1. The previous version of the setup featured two combined with mirrors, to generate six views. The use of mirrors introduced problems with the focus, as the distance for light to travel from the paw through one of the mirrors to the camera could be as much as twice as far as the closest distance to the camera. Additionally, recording speed was increased from 200 frames per second to 300 frames per second, to enable better tracking of the very quick movements of the rats. As no behavioural videos are yet recorded using the new camera setup, the old ones are presented in Figure 2.

D. Calibration

Calibration was performed by placing an object with known geometry at a predefined location and then manually selecting the object corners in each of the videos. Given these selected image points, the camera matrix was computed using direct linear transform (DLT) [2].



(a) (b) (c) (d)

Fig. 2: Video inputs from four views in subfigures (a),(b),(c) and (d) with superimposed projections of pose estimation results. In this particular frame, the paw of the rat is approaching the food pellet through a slit in the wall. The nose can also be seen in this frame. The digits are illustrated in red, green, blue and cyan, where red corresponds to the human index finger, green corresponds to the middle finger, etc. Note that the rat paw only has four long digits; the thumb is considered too short and is not used in this model. The palm of the paw and the forearm are yellow, the nose is black and the food pellet is magenta.

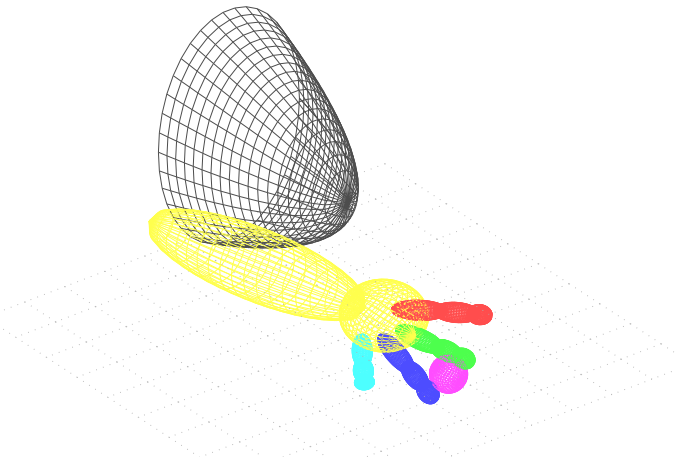


Fig. 3: The model for the paw, nose and pellet, with colors as explained in Fig 2.

II. Tracking

A. Related work

The subject of hand pose estimation and tracking has gained a lot of attention the last 5-10 years as the computational power available is approaching what is needed for real-time applications [3]. In human applications it is possible to use for example colored gloves [4],[5], or using a combination of normal cameras and depth cameras [6]. However, for the presented application this is not possible as the movements are so fast and the paws are so small (less than 1cm in width), and the rats are not as collaborative as humans can be.

B. Overview of the method

The tracking method is generative, i.e. the paw is modelled as explained in section II-C, below, with 22 variable parameters and these parameters are subsequently estimated as the argument of the maximum of the quality function given in section II-G. Evaluation of the quality function is derived starting at the projection of quadrics in section II-D, followed by the evaluation of the similarity of measured images and estimated poses in sections II-E, II-F and II-G. A brief presentation of the algorithm used for maximizing the quality measure in each frame and over time is presented in section II-H.

C. Rat model

The rat paw consists of four long *digits* (”fingers”) and each digit consists of three *phalanges* (bones). The paw is modelled by 13 quadrics, of which 12 represent ellipsoids for the phalanges and one represents an ellipsoid for the palm of the paw. Furthermore, the forearm is modelled as an ellipsoid and the nose is modelled as an elliptic paraboloid. In total, the rat pose is modelled using 15 quadrics, as seen in Figure 3. Due to anatomical constraints, the kinematics of each digit can be modelled using only four degrees of freedom - one for adduction/abduction at the proximal joint and three for flexion at each one of the joints. Consequently, the paw can be modelled using 16 parameters for the digits, four constant vectors representing the metacarpal bones and 6 parameters for position and rotation of the palm of the paw. Furthermore, the forearm is assumed to be fixated at the wrist and can rotate along all three axes in space. This amounts to a total of 22 parameters.

D. Projection of quadrics

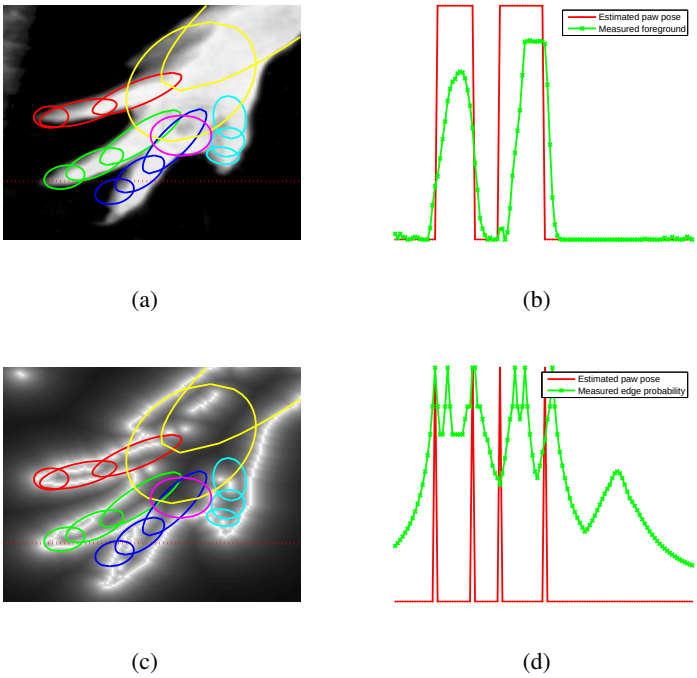
A quadric surface in 3D can be described as the solutions $\mathbf{X} = (X, Y, Z)$ to the quadratic equation

$$[X \quad Y \quad Z \quad 1]\, C \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = 0 \tag{1}$$

The projection of such a surface onto an image plane is a conic section which can be described as the solutions $\mathbf{x} = (x, y)$ to the quadratic equation

$$[x \quad y \quad 1]\, C \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = 0 \tag{2}$$

As described by [7], the conic matrix C can be easily computed along with the depth of any projected point on the quadric.



(a) (b)

(c) (d)

Fig. 4: An image (a) with superimposed projections of pose estimation results and a horizontal dotted line depicting the row of choice for (b). (b) is an illustration of the row shown in (a), where the red line shows where the image is covered by the estimated paw pose and the green line is the grayscale values of the row. (c) is an illustration of the edge-probability used in (7) and (d) is, analogously to (b), an illustration of the particular row of choise in (c).

E. The quality function

The quality function used here is a measure of how well an estimated pose fits the intensity and edges of an image. The area quality function for one view is defined as

$$q_A = \frac{\sum_{x,y} \min(P(x,y), F(x,y))}{\sum_{x,y} \max(P(x,y), F(x,y))} \tag{3}$$

where P is the projection of the estimated paw pose and F is the measured foreground image. Here P is binary while F is normally not. Note that for binary functions, the quality function is identical to the Jaccard index, which is used in [1]. The measured foreground image F is defined as the image minus the background, i.e. $F = I - B$. The projection of the paw pose, P, is defined as the binary image with 1's at each pixel covered by a projected quadric of the estimated pose, and 0's elsewhere. Thus, we are interested in evaluating how much of the projection of a quadric covers the foreground of an image, i.e. to compute the sum of the elements in an image that are covered by the filled conic. The naive way to evaluate which pixels are covered by a conic is to evaluate (2) for each pixel, which would require MN evaluations of (2). This is the method used in [1] and is very time consuming. This can be improved by noting that for each fixed $x_0 \in [1, M]$,

$$\begin{bmatrix} x_0 & y & 1 \end{bmatrix} C \begin{bmatrix} x_0 \\ y \\ 1 \end{bmatrix} = 0, \tag{4}$$

is a quadratic equation in y. Then the line segment $\{y_0\} \times [y_0, y_1]$, where y_0 and y_1 are solutions to (4), covers the conic. Each such slice can be computed by finding the real roots of the quadratic equation. If there are no real roots, the conic does not intersect that particular row, x_0. As the estimated paw pose consists of 16 quadrics, for each row x_0 there is a (potentially empty) set of intervals that cover parts of the row. After merging overlapping intervals, we get a set of disjoint intervals for each row. Computing the contribution to (3) from one such interval, can be done very efficiently using integral images. Let $I_F(x,y)$ be the integral foreground image

$$I_F(x,y) = \sum_{j=0}^{y} F(x,j) \tag{5}$$

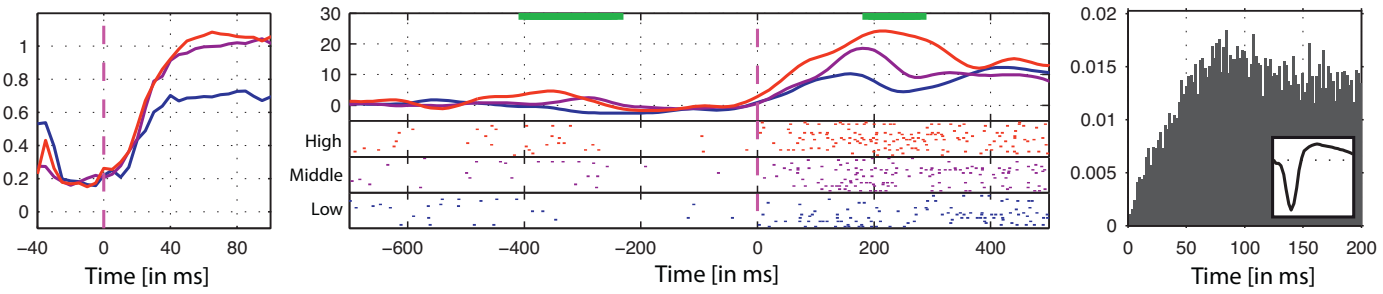
Then the contribution to (3) from an interval $[y_0, y_1]$ is given by

$$I_F(x,y_1) - I_F(x,y_0) \tag{6}$$

Thus the cover of a projected quadric can be computed in linear time (with respect to the smallest side of an image). For each row, 16 equations equations need to be solved, up to 16 intervals merged into disjoint intervals and grayscale values of the integral images at the endpoints of each interval needs to be evaluated. This is considerably faster than naive evaluation of (3).

F. Edge quality function

Let D be a matrix where, for each coordinate (x,y), $D(x,y)$ is the shortest distance from (x,y) to an edge in an image I (can be computed efficiently using distance transforms). Then for each point (x,y) on the edge of the projected



(a) (b) (c)

Fig. 5: Neural correlation to motor components. Examples of neurons correlated to individual motor components. Reach attempts are sorted depending on the peak value for a motor component during the attempt, as shown in (a). (b) is sorted for grasp so that the lowest degree of fist closure are pooled into the blue group, the intermediate closures into the purple group and the highest into the red group. (b) show standardized peri-event firing rates and raster plots of individual trials for each group, aligned to time point of maximum paw extension and averaged over all attempts in each group. A green line indicates a significant difference in firing rates between any of the three groups ($p < 0.05$). (c) shows the autocorrelogram and waveform for each presented neuron. Modified from [1].

paw pose, $\frac{1}{D(x,y)+1}$ is related to the probability of (x, y) corresponding to an edge. The edge quality function is defined as

$$q_E = \frac{1}{|\partial P|} \sum_{(x,y)\in\partial P} \frac{1}{D(x,y)+1}. \tag{7}$$

G. The quality function

For a predicted paw pose, a point in time and each camera, the quality function is defined as (3). The quality in each of the views are multiplied to give the quality function for a predicted pose and a point in time:

$$q = \prod_{k=1}^{N} \left(q_A^{(k)} \lambda + (1-\lambda) q_E^{(k)} \right) \tag{8}$$

H. Maximizing the quality function

As the movement of the paw was so quick with respect to the recording speed, using temporal consistency to improve tracking proved to be difficult. Thus the method presented here is only based on frame-by-frame pose estimation. For each frame, a number of previously encountered poses together with the pose of the previous frame was evaluated, to find an appropriate starting point. Then a gradient-descent like optimization method was applied to find a local maxima, which given that the more coarse global optimization is close to the global maxima, is also the global maxima.

III. Alignment of neural recordings to kinematic data

Nerve cells in the brain communicate by briefly altering their membrane potential, these impulses are referred to as action potentials. Already in the first analyses of the changes in neuronal activity patterns related to different aspects of the motor output, we could identify single nerve cells that appeared to show specific modulations in the frequency of emitted action potentials in relation to specific parts of the motor sequence. In Figure 5, the recorded neuronal activity of a nerve cell, in multiple reaching trials, was aligned to the part of the reaching movement where grasping is initiated. It was noted that the average number of action potentials detected scaled with the degree of fist closure suggesting that this nerve cell may have a role in controlling this part of the reaching and grasping compound movement.

IV. Discussion

The presented system uses a calibration procedure (DLT) that is easy to code but is numerically unstable and is a potential bottleneck for the rest of the system. Even though calibration quality has been improved by using views with higher resolution, we aim to improve this part by using a new calibration method that will also be easier to perform during the experiments.

References

[1] T. Palmér, M. Tamtè, P. Halje, O. Enqvist, and P. Petersson, "A system for automated tracking of motor components in neurophysiological research," *Journal of Neuroscience Methods*, vol. 205, no. 2, pp. 334–344, 2012.

[2] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004, second Edition.

[3] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly, "Vision-based hand pose estimation: A review," *Computer Vision and Image Understanding*, vol. 108, no. 1, pp. 52–73, 2007.

[4] J. Usabiaga, A. Erol, G. Bebis, R. Boyle, and X. Twombly, "Global hand pose estimation by multiple camera ellipse tracking," *Machine Vision and Applications*, vol. 21, no. 1, pp. 1–15, 2009.

[5] R. Y. Wang and J. Popović, "Real-time hand-tracking with a color glove," in *ACM Transactions on Graphics (TOG)*, vol. 28, no. 3. ACM, 2009, p. 63.

[6] M.-F. Ho, C.-Y. Tseng, C.-C. Lien, and C.-L. Huang, "A multi-view vision-based hand motion capturing system," *Pattern Recognition*, vol. 44, no. 2, pp. 443–453, 2011.

[7] B. Stenger, P. R. Mendonça, and R. Cipolla, "Model-based 3d tracking of an articulated hand," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 2. IEEE, 2001, pp. II–310.