CENTER FOR
MACHINE PERCEPTION

CZECH TECHNICAL
UNIVERSITY IN PRAGUE

TECHNICAL REPORT

Reference Tracking System for a Mobile Skid Steer Robot

(Version 1.00)

Vladimír Kubelka, Vladimír Burian,
Přemysl Kafka

kubelka.vladimir@fel.cvut.cz

May 9, 2012

Available at
https://cw.felk.cvut.cz/doku.php/misc/projects/nifti/sw/start

Center for Machine Perception, Department of Cybernetics
Faculty of Electrical Engineering, Czech Technical University
Technická 2, 166 27 Prague 6, Czech Republic
fax +420 2 2435 7385, phone +420 2 2435 7637, www: http://cmp.felk.cvut.cz

Reference Tracking System for a Mobile Skid Steer Robot

Vladimír Kubelka, Vladimír Burian, Přemysl Kafka

May 9, 2012

Abstract

The reference tracking system was developed for the NiFTI robot odometry system performance verification as a "Prace v tymu a jeji organizace A3M99PTO" course semester project. It uses a single video camera to track the robot movement in a plane. A distinct colored marker must be attached to the robot to be tracked. The tracker can also determine the robot azimuth given the target contains two differently colored areas. This report explains the technical background of the reference system including the tracking algorithm, transformations and synchronization. The accuracy of the system is discussed as well.

NIFTi Natural Human-Robot Cooperation in Dynamic Environments

Contents

1 Introduction

The aim of the project was to design and create a video camera reference tracking system. The system would determine position and heading of a mobile robot using image processing methods. The specifications were:

- an easy-to-install camera setup
- usable outdoor, independent on light conditions
- supporting the robot odometry system including synchronization

All the specifications were fulfilled reaching localization accuracy (15 ± 13) cm and heading accuracy (3.8 ± 2.7) degrees (the heading accuracy depends on the robot distance, the position accuracy applies for the whole experiment area).

2 The reference tracking system structure

The components, described in the next sections, cooperate as follows:

1. The recorded video is tracked using our colored-target-based tracker. Several approaches have been tested, however, this one shows the most accurate and stable results.

2. The tracked path expressed in image coordinates is transformed into world coordinates using a homography transformation based on calibration pairs recorded in the beginning of an experiment.

3. The transformed path is synchronized with the odometry path, so the resulting track and odometry vectors have same number of samples and cover the same time period.

3 Detailed description of the main components

3.1 Homography transformation

The tracked coordinates are expressed in the camera image coordinates. The task is to express these coordinates in the world frame. Using a single camera, it is possible if the robot moves in a plane. This relation is called homography (Fig.: 1). The relation can be expressed using homogeneous coordinates as

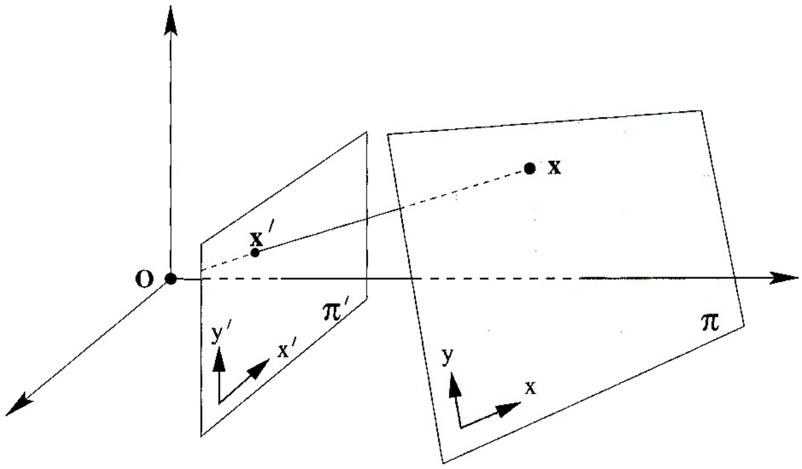


Figure 1: Homography: projection from one plane to another *(Multiple View Geometry; Hartley, R and Zisserman, A; Cambridge University Press; 2003; p.34)*

$$a \begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \end{bmatrix} \tag{1}$$

or more briefly $x' = Hx$. The x vector contains the image coordinates $[x_1, x_2, 1]'$ and the result of the equation $x' = [x'_1 x'_2 w']'$ contains the world coordinates $\left[\frac{x'_1}{w} \frac{x'_2}{w}\right]$, a stands for a scale.

The matrix H defines the homography up to the scale, thus, we have to evaluate its 8 elements. To do that, calibration pairs of corresponding points are collected (at least four points, from which no three lay in a line) and using the Singular Value Decomposition, the best solution is found. The algorithm accepts 4 or more pairs, over-determining the equation increases accuracy of the transformation. The pairs are selected in the recorded video and appropriate world coordinates are entered.

3.2 Synchronization

The samples of the tracked path are described by a number of a frame, the frame rate depends on the video camera, it is most likely 30 frames per second. Coordinates originating in the robot odometry system are sampled with frequency 100 Hz. It is impossible to start the odometry system and video recording perfectly synchronously, therefore, these two vectors are shifted in the absolute time. To synchronize them, several steps are performed:

1. The user marks two corresponding moments in the both signals (Fig.: 2)
2. Based on these moments, a time vector for the tracked coordinate vector expressed in the robot odometry time is evaluated
3. For each odometry sample, a reference value is interpolated from the tracked vector using the new time vector
4. The longest overlay of these two vectors is found and based on it, both vectors are cropped.

The result is a vector of the odometry system samples and corresponding reference vector having the same number of samples.

3.3 Tracker

Principle of the histogram tracker operation is roughly following:

1. It's created a histogram of all three components in HSV color space of pixels in area we want to track. And around the chosen area is created tracking window.
2. Pixels in movie frame are rated on a basis of histogram model evaluated in the first step.
3. Center of mass of pixels inside the tracking window is evaluated. Weight is a pixel's rate. Center of mass coordinates are the tracker's result in actual frame.
4. Tracking window is moved to the newly computed center of mass.
5. Next movie frame is read and process continues from step 2.

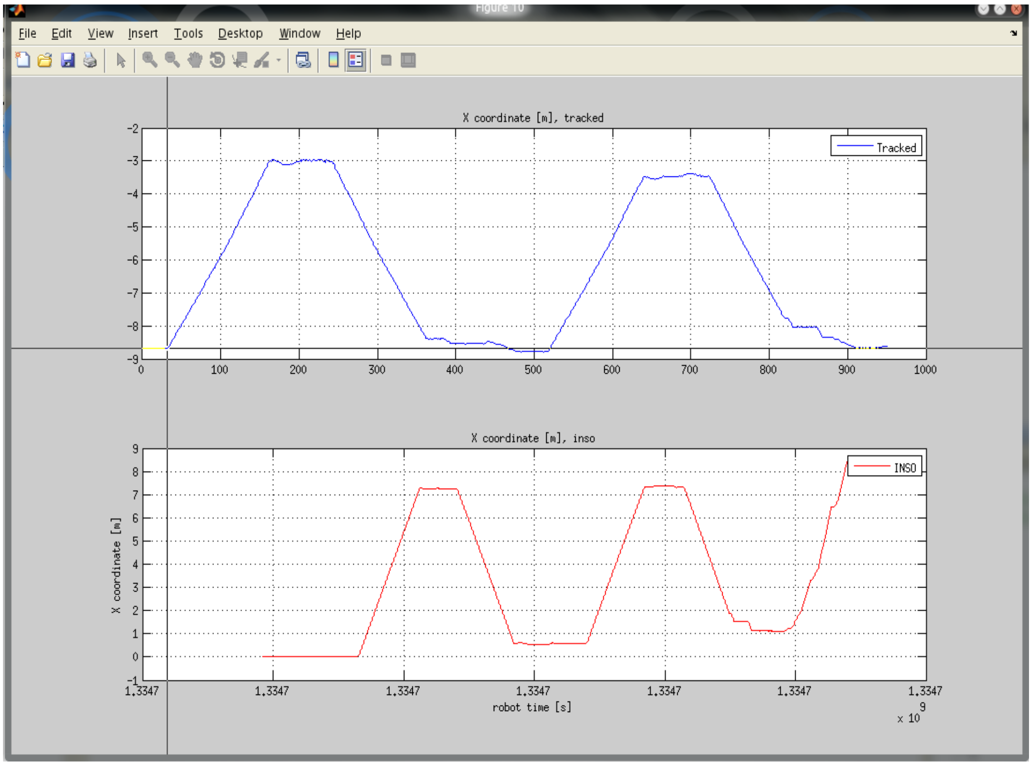


Figure 2: Marking the beginning of the experiment in both signal (odometry and reference)

Example of a histogram model created from a green square placed on the robot (figure 4) can be seen in figure 3. Before histogram computation, noise in RGB and HSV color space is added to the pixels taken from the selected area. It's the reason of the histogram balance. And so the histogram model represents noise present in images and slight changes in object color due to e.g. a change in a lightening.

Pixels rating evaluation proceed by reading a value from the histogram model in a point of the pixel value in HSV space. The movie frame after this step can be seen in figure 5. The figure also illustrates that only the hue component of HSV space is insufficient to rate pixels successfully. It's due to the noise in hue added probably by a video codec. That's why the pixels are rated on a basis of all components in HSV space.

Coordinates of the object being tracked are computed as a center of mass of rated pixels inside the tracking window. Tracking window is round, it's radius is evaluated as a double of size of the area selected at the beginning. It's important that the window is so big so the whole tracked object is not moved out of the window between movie frame step. Otherwise there is

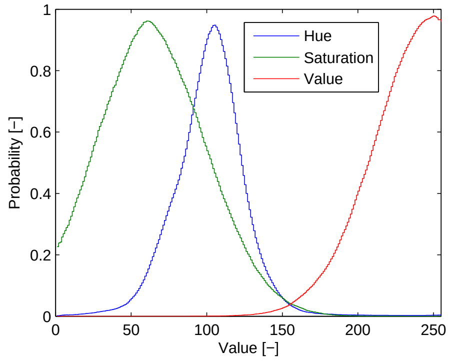


Figure 3: Example of histogram model created from a green square placed on the robot.

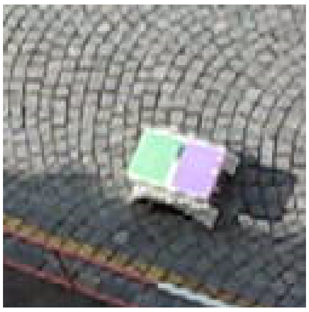
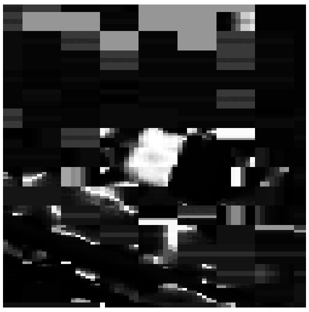
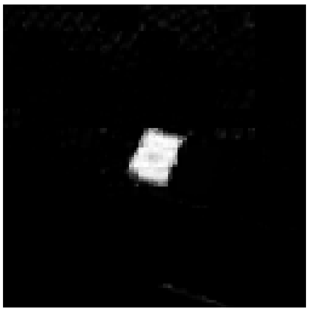


Figure 4: Movie frame with the robot taken from a window.



(a) Rate by color



(b) Rate by all three HSV components

Figure 5: Pixels rating in a movie frame evaluated on a basis of a histogram model.

systematic error present. This condition is not hard to comply as the robot is moving very slowly. In the other case the robot position would be predicted by the Kalman filter. Concurrently the window should be as small as possible so the disturbing pixels are kept outside.

In the worst case it's possible to suppress the disturbing pixels by setting lower threshold in code. Or possibly by reduction of the noise added during the histogram model creation. It's also possible to enable experimental functions of adaptive threshold and adaptive window size. Adaptive threshold can efficiently suppress disturbing pixels and make the tracker much more accurate sometimes, but sometimes the opposite can happen. Adaptive window size is currently troubleshooting as the window size is very expansive.

4 Accuracy

4.1 Requirements

One of the requirements for the reference system was accuracy. It was was not exactly specified, but logically, a reference system should be significantly more accurate than the system that uses it as a reference. Compared to the present robot odometry system, this requirement was fulfilled. The reference system reaches accuracy (15 ± 13) cm per 17m (0.8%). Given the outdoor environment, the robot odometry system performance is significantly worse.

The reference system determines the robot heading as well. Accuracy of this value strongly depends on the tracked target distance from the camera. The best results achieved were (3.8 ± 2.7) degrees, the worst (22.7 ± 9.8) degrees, as will be shown below.

4.2 Position accuracy evaluation

To verify the position accuracy of the reference system, the robot was navigated over a series of points (Fig.: 6), which coordinates were measured as exactly as possible. The outcome of the reference system was subsequently compared to these measured coordinates (Fig.: 7) and the mean error was determined. Since there were some obviously systematic errors, the best value and the worst were discarded and the previously mentioned value (15 ± 13) cm resulted.

4.3 Heading accuracy evaluation

To verify the heading accuracy, a similar approach was chosen. The robot was equipped with a special target with two areas of a different color. The robot was navigated over a rectangle and its diagonal(Fig.: 8). Comparing the known heading angles from the geometry and the tracked ones, accuracy values were evaluated for edges of the rectangle (Fig.: 9). From its development, it can be assumed, that further the colored target gets from the camera, the worst result we get. However, it is necessary to mention the alternative colored target we have prepared but did not manage to test. We expect better results with that one.

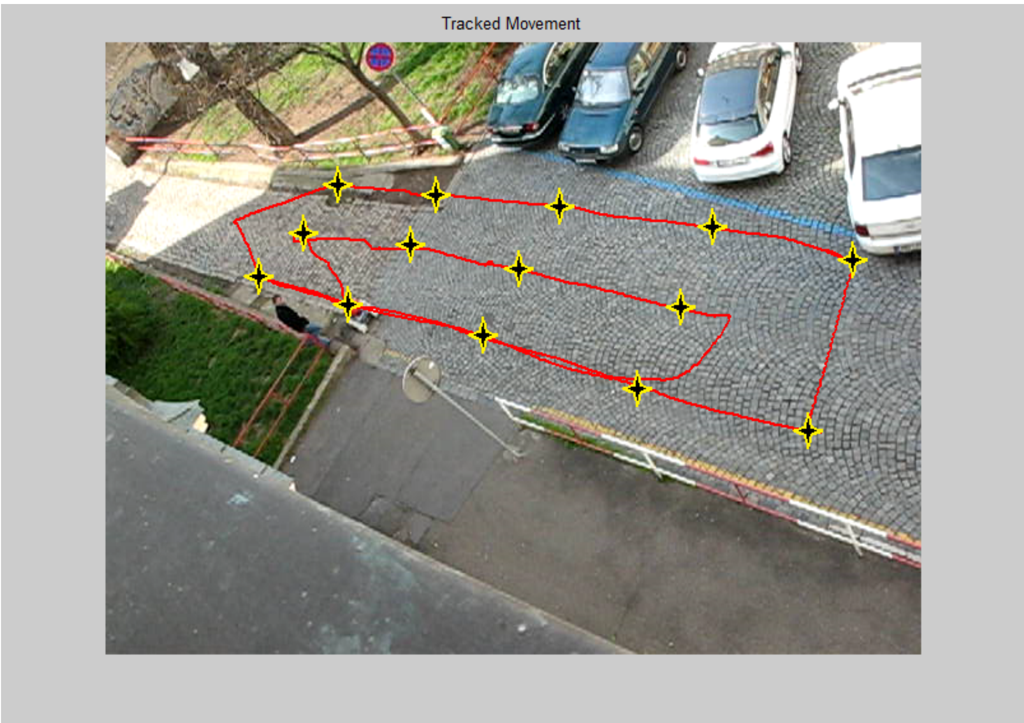


Figure 6: A grid of points with precisely known coordinates

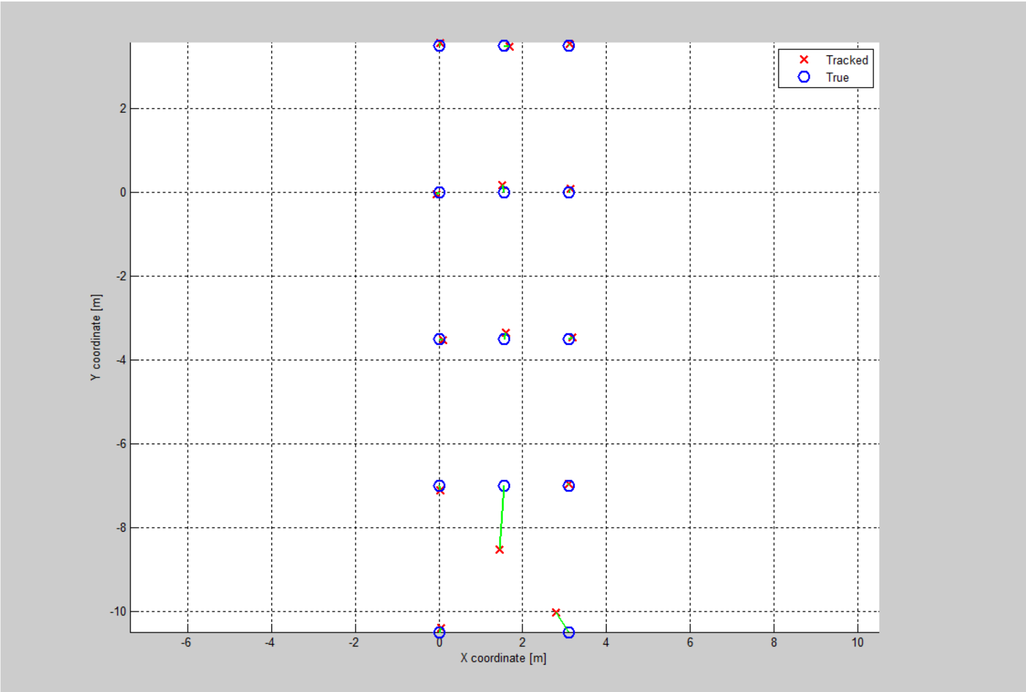


Figure 7: Comparison of the tracked and measured values

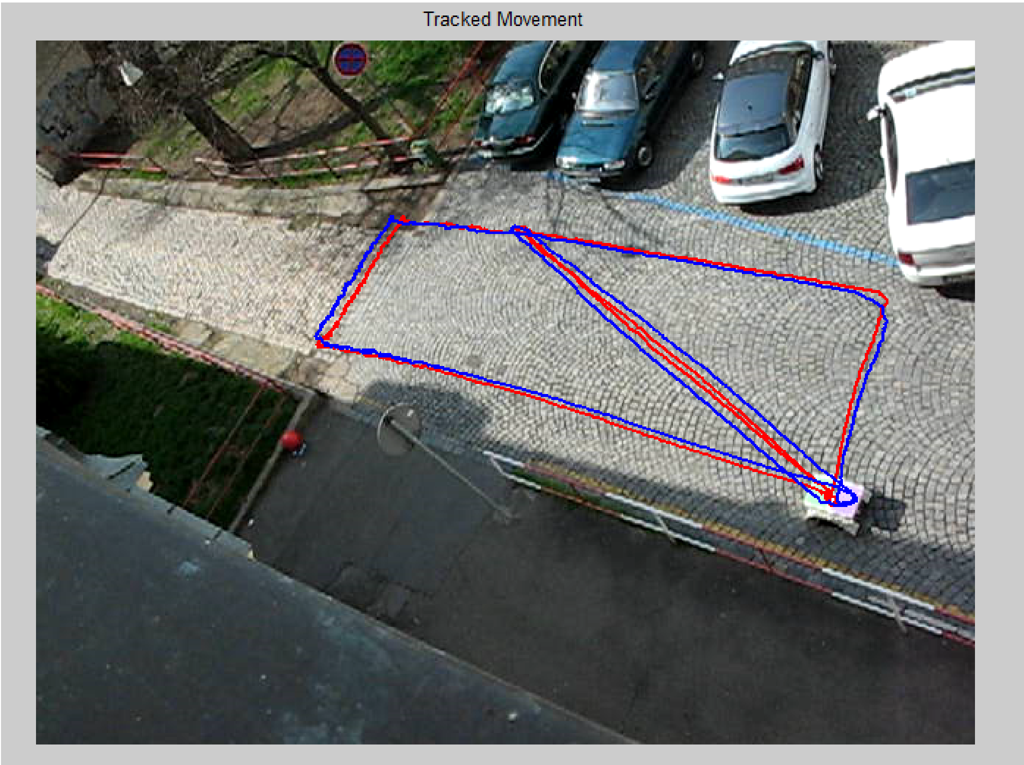


Figure 8: A rectangle path with one diagonal line

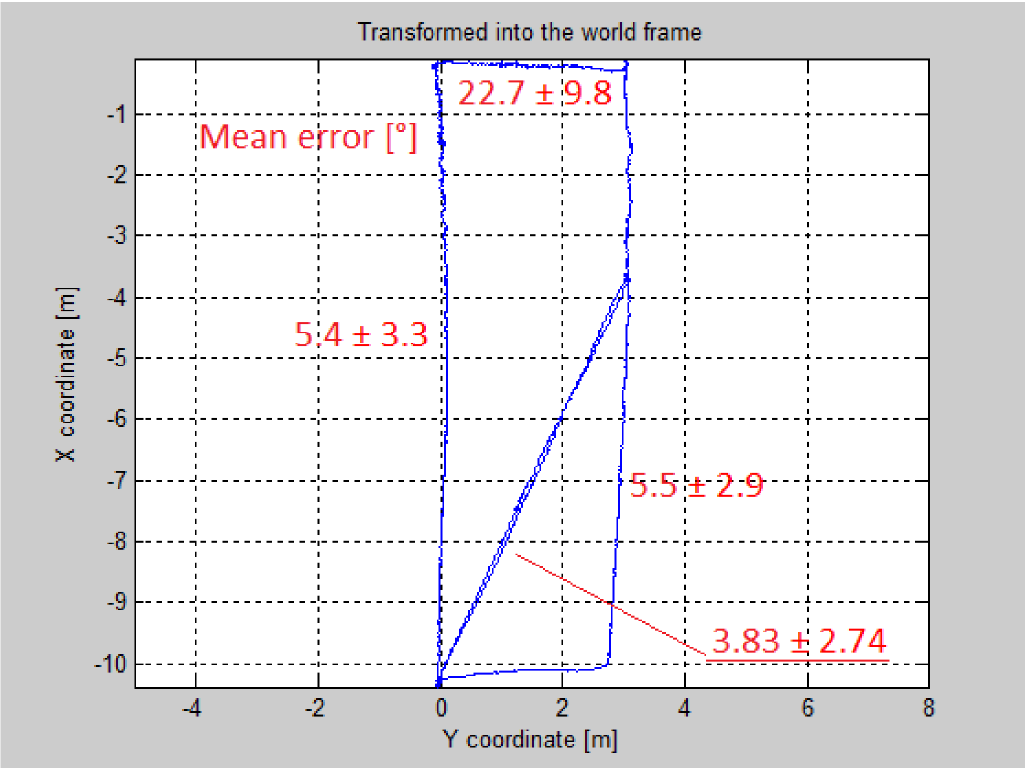


Figure 9: Heading angle error expressed for distinct edges of the rectangle