Quantitative Bilateral Symmetry Detection for Robust Illumination Invariant Features

Kai Hübner (Uni Bremen), Daniel Westhoff (Uni Hamburg)

email: westhoff@informatik.uni-hamburg.de

Introduction

In this work, we propose a novel method for the determination of illumination-invariant features in images. Quantitative bilateral symmetry of a given scene is computed using *Dynamic Programming*. Compared to other approaches for symmetry detection, results of our novel algorithm describe symmetry in terms of an absolute region instead of a relative degree. Here, symmetry axes are extracted using vertical symmetry images, non-maxima suppression and hysteresis thresholding. For each symmetry image a unique feature vector is obtained as the sum of the gray-values of each column of the image. In the experiment, the feature vector is used to robustly track motion in an image sequence.

Dynamic Programming Symmetry (DPS)

The ideas of Dynamic Programming are picked up to implement a new approach to symmetry detection referred to as Dynamic Programming Symmetry (*DPS*) algorithm. In the case of symmetry detection, patterns derive from one and the same image. The measure of correspondence between these patterns is found using Dynamic Programming.

The result of the Dynamic Programming technique is seen as a the symmetry value of one currently treated image point depending on the symmetric properties of its two surrounding patterns. In Fig. 1, the resulting costs and the optimal path are presented for two asymmetric example patterns using weight parameters $W_i = 1$.

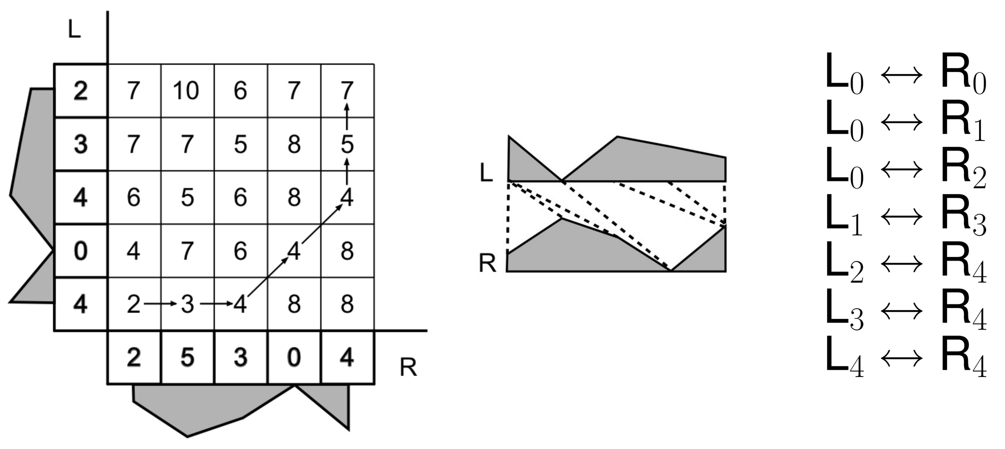


Figure 1: Asymmetric shapes: Complete search space including costs and corresponding mapping path ($W_i = 1$).

Note that an optimal path can always be found, while only the shape and the resulting cost of the path inform about the quality of symmetry. In Fig. 1, a symmetrical similarity can be found by compression of the patterns, but only through a serious transformation of both structures. The cost for this modification can be observed by means of the error measures on the optimal path. An exact mapping $\forall i : L_i \leftrightarrow R_i$ with unanimous overlay with small error results after inverting R, as is shown in Fig. 2:

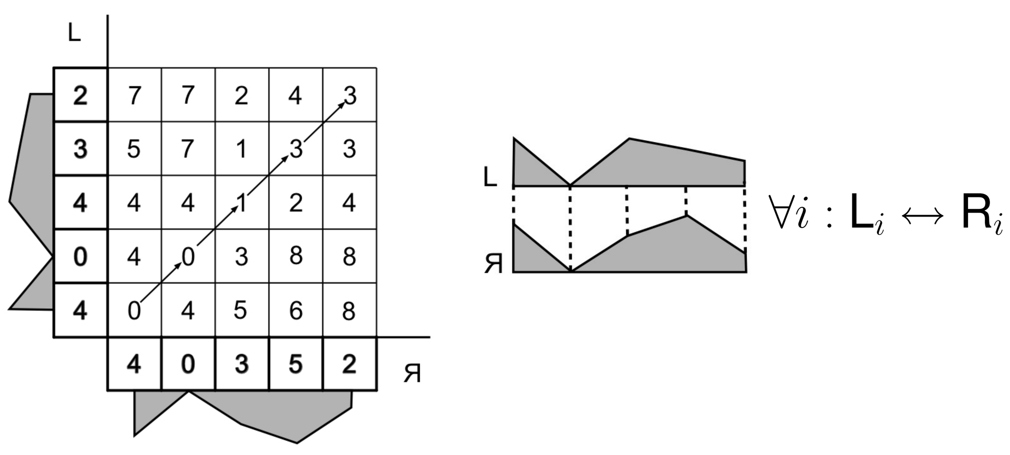


Figure 2: Symmetric shapes: Complete search space including costs and corresponding mapping path ($W_i = 1$).

Up to this point, the Dynamic Programming method is based on a fixed search space square over the complete patterns. An indexing for computing the cells proves not only to be useful but also efficient to iteratively give evidence about symmetry in a certain subsquare. Path and failure resulting from a subsquare of side length m describe the symmetric properties of the patterns in a subregion of size m. To return a symmetric value or range as a result, heuristics are required to continue or to break this iteration.

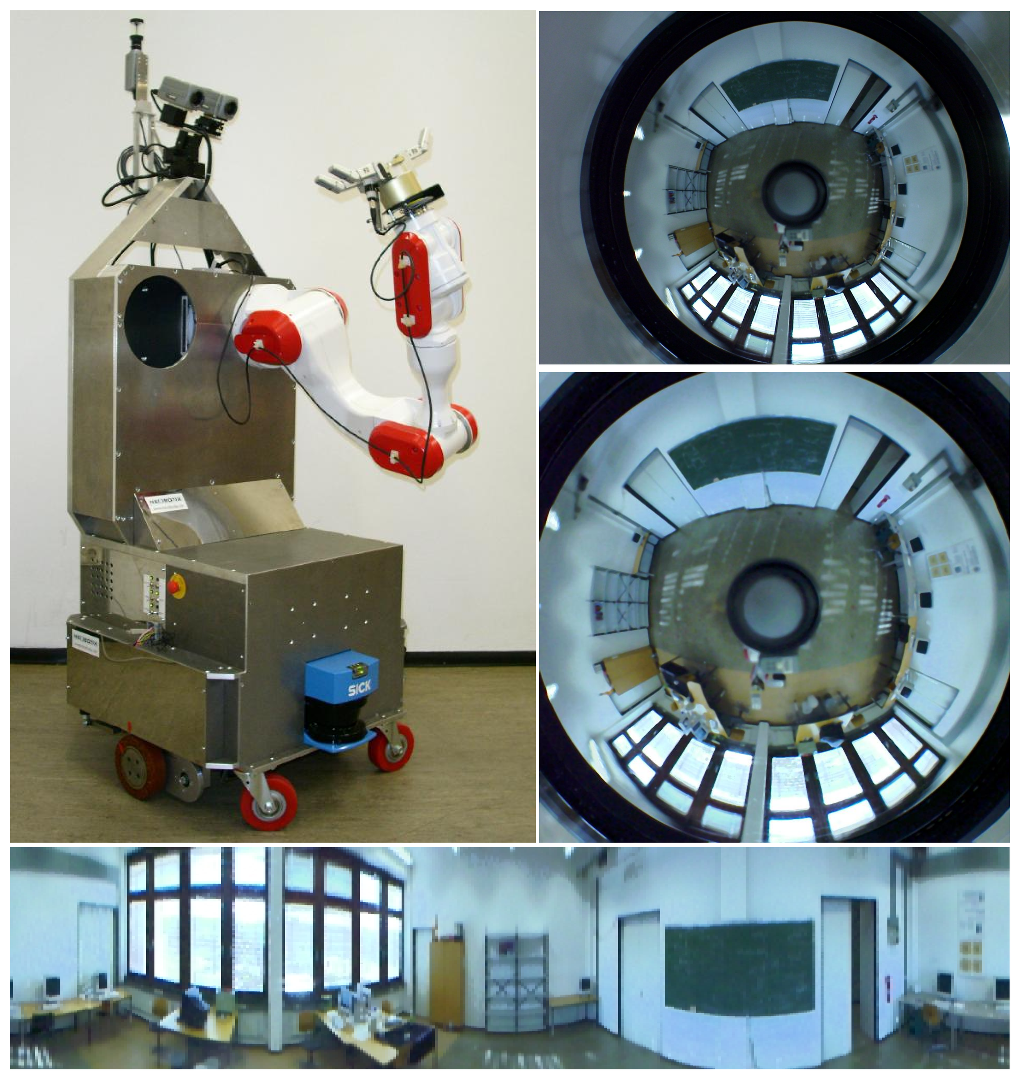


Figure 3: Generating a panoramic image.

An acceptable course of action for that purpose is the choice of a failure threshold on the optimal path.

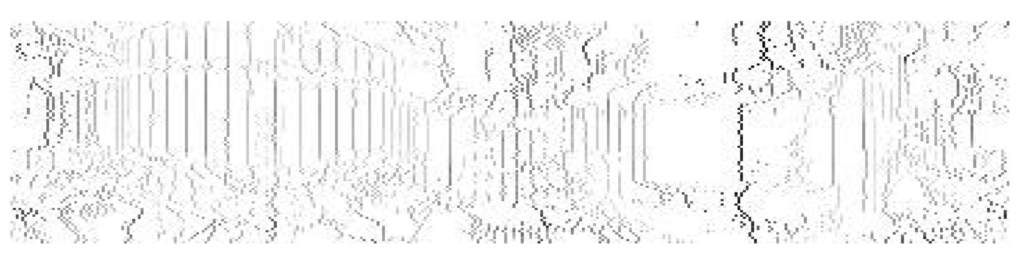
Symmetry Image Signatures

Image signatures are applied in applications like image retrieval or mobile robot localization. An image signature is a vector that describes an image unique. We propose to compute an image signature using gray-value symmetry images generated by the DPS algorithm. The examples presented are panoramic images taken from an omnidirectional vision system mounted on the mobile robot TASER (see Fig. 3).

The omnidirectional vision system uses a hyperboloidal mirror. Before the omnidirectional images from the camera system are transformed into panoramas, the images are preprocessed with a median filter and a gray-value histogram equalization to improve the contrast. The panoramic transformation is done by hyperboloidal projection as proposed by Yamazawa *et.al.* in 1993. The pre-processing steps are shown in Fig. 3. After the pre-processing of an image, the symmetry signature is computed with a four-step algorithm:

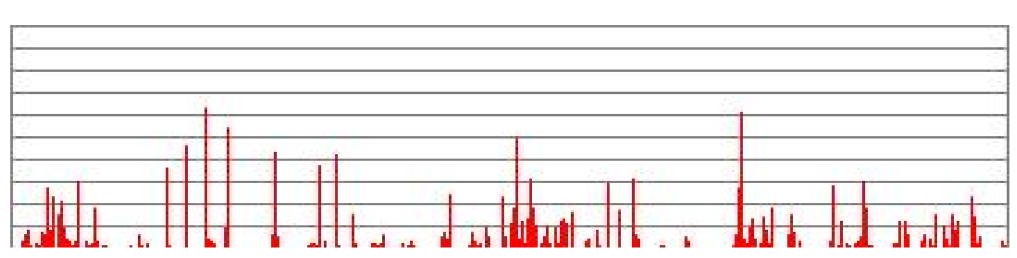
1. The DPS algorithm is used to create a gray-value image indicating vertical reflective symmetry.

2. A non-maxima suppression is used for each row to extract vertical symmetry axes.



3. A hysteresis threshold can be used to eliminate pixels that represent small symmetric areas.

4. A symmetry signature is created by summing up all gray-values of each individual column of the image.



The algorithm takes advantage of the fact that panoramas are cylindrical images where the vertical borders can be extended by a copy of the image itself. Because of this, correct quantitative symmetry values are detected near these image borders.

Since symmetry is robust against illumination changes, the signatures of images of the same place taken under different illumination conditions vary only very little.

Experiments

To demonstrate the usability of symmetry image signatures, some preliminary experiments have been accomplished. The underlying idea is to find a region of interest for the service robot of Fig. 3. Detecting motion in a panoramic image taken from the omnidirectional vision system will help the robot to determine where it may find people.

The robot takes a panoramic image of its empty surroundings and computes the symmetry image signature. We will refer to this vector as the reference signature $\vec{s}_{ref}$. If the images are noisy, i.e. due to poor illumination conditions, more than one image may be taken and the reference signature is the mean vector of all signatures. After the reference signature has been determined, images from the omnidirectional vision system are taken and their symmetry signature $\vec{s}$ is compared to the reference signature. Therefore, the components of an error vector $\vec{e}$ are computed using the following equation:

$$e^i = (s^i_{ref} - s^i)^2. \qquad (1)$$

If the environment of the robot changes, the symmetry will change in the area where the motion takes place. The components of the error vector for the respective columns now carry high error values.

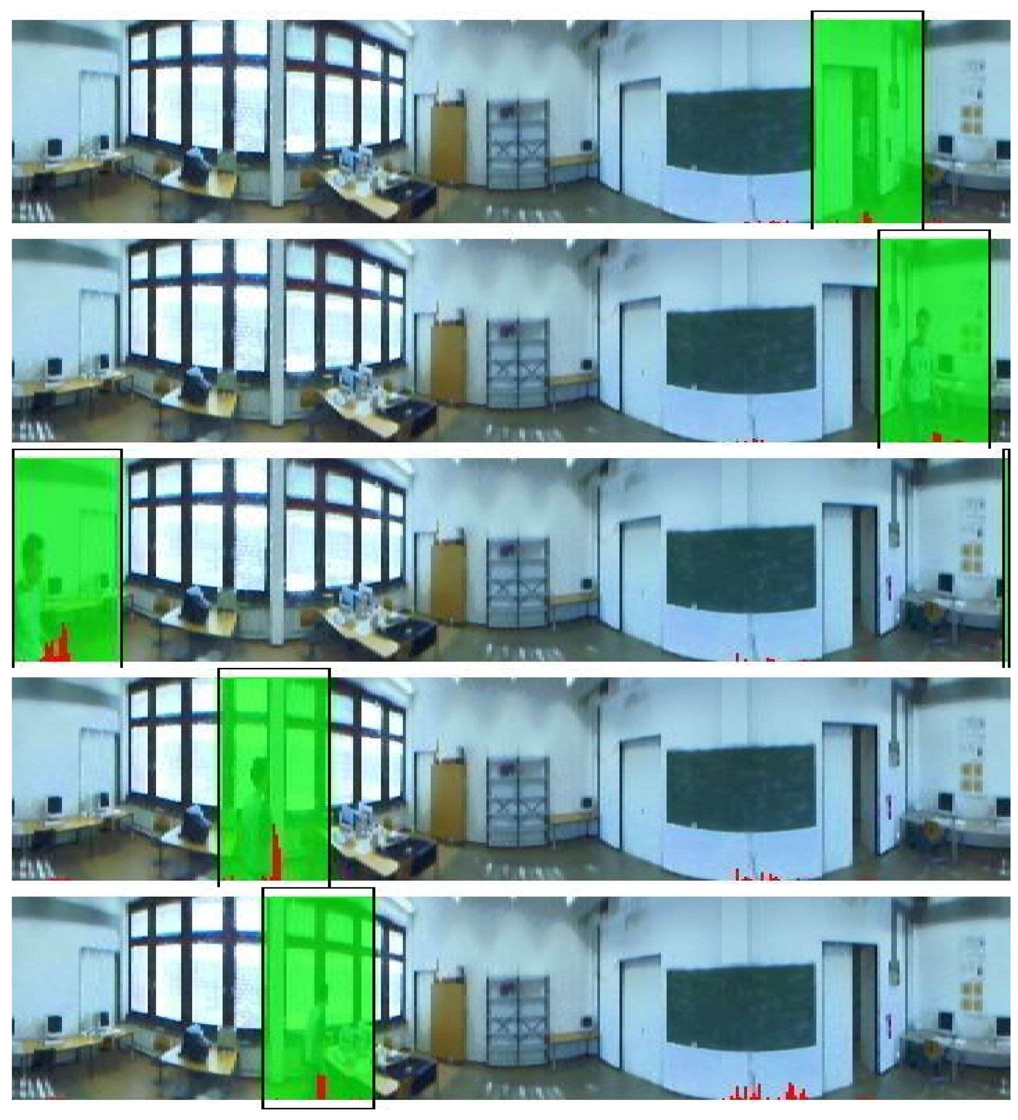


Figure 8: Examples from an tracking sequence.

Fig. 8 shows some images from an image sequence where the error vector is visualized by bars at the bottom of each image. Around the image column with the highest error value a region of interest has been marked. The region of interest may now be used for more elaborate tracking methods or otherwise to turn the robot into the direction of motion, since each column of the panorama image corresponds to an orientation between $-180°$ and $180°$.

Very large vertical symmetry axes may produce high error values because of noise, e.g. see Fig. 8 at the center column of the blackboard. This leads to error values exceeding those for columns where motion takes place. In this case, the region of interest sometimes skips for singular frames to these highly symmetric columns.