14

Navigation in a Box Stereovision for Industry Automation

Giacomo Spampinato, Jörgen Lidholm, Fredrik Ekstrand, Carl Ahlberg, Lars Asplund and Mikael Ekström
School of innovation, design and technology, Mälardalen University
Sweden

1. Introduction

The research presented addresses the emerging topic of AGVs (Automated Guided Vehicles) specifically related to industrial sites. The work presented has been carried out in the frame of the MALTA project (Multiple Autonomous forklifts for Loading and Transportation Applications), a joint research project between industry and university, funded by the European Regional Development and Robotdalen, in partnership with the Swedish Knowledge Foundation. The project objective is to create fully autonomous forklift trucks for paper reel handling. The result is expected to be of general benefit for industries that use forklift trucks in their material handling through higher operating efficiency and better flexibility with reduced risk for accidents and handling damages than if only manual forklift trucks are used.
A brief overview of the state of the art in AGVs will be reported in order to better understand the new challenges and technologies. Among the emerging technologies used for vehicle automation, vision is one of the most promising in terms of versatility and efficiency, with a high potential to drastically reduce the costs.

2. AGVs for industry, new challenges and technologies

Commonly known as AGVs, automatic vehicles able to drive autonomously while transporting materials and goods are present on the market since the middle of the 20th century. They are both used in indoor and outdoor environments for industrial as well as for service applications for improving the production efficiency and reducing the staff costs.
In field robotics, fully autonomous vehicles are of great interest and still a challenge for researchers and industrial entrepreneurs. The concepts of mobile robotics in indoor and outdoor environment has already exploded on the market in the recent years with a large amount of "intelligent" products like autonomous lawn mower, vacuum cleaner robots, and ATS (Automatic Transportation Systems) in public services. Although the huge amount of automatic moving platforms already present on the market, almost no one is able to perform automatic navigation in dynamic environments without predefined information. In indoor environments, traditional AGVs typically rely on magnetic wires placed on the ground or other kind of additional infrastructures, like active inductive elements and reflective bars, located in strategic positions of the working area. Such techniques are mostly used by AGVs

to provide autonomous transportations in industrial sites, (Danaher Motion, Corecon, Omnitech robotic, Egemin Automation), and in service environments like hospitals (TransCar AGV by Swisslog, ALTIS by FMC and MLR). These systems mainly rely on bi-dimensional views from conventional laser based sensors and need pre-defined maps of the environment. As a consequence they show very low flexibility to environment changes.
On the other hand, in outdoor environments, AGVs mainly rely on high-precision global positioning systems (GPS) and predefined maps. One classical example is represented by the construction vehicles field. The current generation of autonomous hauler trucks (the Front Runner system from Komatsu shown in Fig. 1), consists of a vehicle controller over a wireless network, operated via a supervisory computer. Information on target course and speed is sent wirelessly from the supervisory computer, while the GPS is used to obtain the position. The architecture is rather classical for outdoor robotics navigation, and makes use of conventional sensors that are costly and do not provide complete 3D information about shapes and obstacles. Moreover, the navigation quality is strongly dependent on the GPS precision, and cannot be used in indoor environments or near buildings. There exist also several mining loaders on the market, from different manufacturers, like Atlas Copco, Sandvik, and Caterpillar that are semiautonomous with very simple trajectory following techniques, and normally remote controlled while loading and unloading.

Fig. 1. Two examples of ATS: the automatic hauler track Front Runner from Kumatsu operating in outdoor, and the autonomous trailer drive from Swisslog operating indoor.

Fully autonomous navigation is still on the research level and rare examples are present on the market as a commercial product. It requires autonomous self localization and simultaneous map building of unknown environments, with additional capabilities of unforeseen obstacles detection and avoidance. Dynamic path planning and online trajectory generation is also essential to guarantee an acceptable trade-off between efficiency and safety. A recent overview of the challenges in dynamic environments can be found in [Laugier & Chatilla, 2007].
On the other hand, vision is broadly recognized as the most versatile sensor for recognition and surveillance in non controlled situations, where the conventional laser based solutions are not suitable without using costly and complex equipments. Typically, 2D environmental representations provided by laser scanners cannot capture the complexity of the unstructured dynamic environments, especially in outdoor scenarios.
At present, industrial vision systems are equipped with fast image processing algorithms and highly descriptive feature detectors that provide impressive performances in highly

controlled situations. However it is not always possible to achieve an adequate level of control of the environmental settings.
Autonomous vehicle navigation is often achieved by using specific infrastructures, which are seen by the system as artificial landmarks. Some examples available on the market are given through video based solutions that use image processing for recognizing different unique patterns spread in strategic positions of the environment (like Sky-trax).
The obvious drawback with this approach is the additional effort required to "dress" the working space with external material not related to the production lines. Sometimes it is also impossible to modify the environmental setting due to the highly dynamic conditions in the production operations.
To overcome these drawbacks, a more versatile and robust vision system is required, which allows automatic vehicle navigation using only pre-existing information from the working setting that is seen by the system as "natural landmarks". This concept requires a new paradigm for the traditional image processing approach that shifts the attention from the two dimensions to the more complete and emerging 3D vision. A three dimensional representation of the operational space is necessary, and modern cameras are able to provide high resolution images with high frame rates. Stereovision is one of the most advanced methodologies today established in the field of 3D vision and utilize the sense of depth and the possibility to build a 3D map of the explored environment by the use of multiple views of the scene. We propose high speed stereo vision to achieve unmanned transportation in structured dynamic environments.

3. Description of the system

The stereo vision system is made of two 5-megapixel CMOS digital image sensors from Micron (MT9P031) and a Xilinx Virtex II XC2V8000 eight million gates equivalent FPGA .

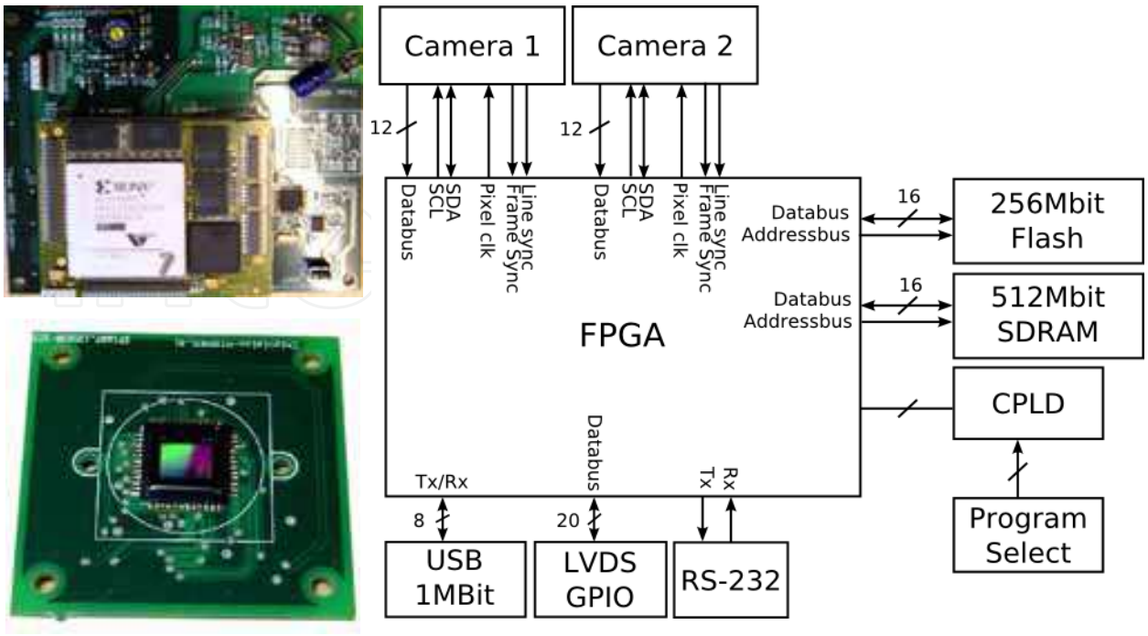


Fig. 2. The HW platform block diagram

On the board there is also 512 MB SDRAM and 256 MB Flash EPROM. The board can hold up to seven different configurations for the FPGA stored in Flash, thus seven different algorithms can be selected during run-time. The FPGA can communicate with the external systems over USB at 1 MBit/ s. The system architecture is shown in Fig. 2.
The final configuration of the stereo system includes the camera sensors, two optical lenses, camera board, FPGA, the power supply and USB interface. All is packed in a compact aluminium box (19x12x4cm3) easy to install and configure through the USB connection. Fig. 3 shows the box and the optics adopted. The lenses adopted are two fisheye lenses with 2,1mm focal length from Mini-Objektiv, with F2.0 aperture and 100 degrees field of view. The system also includes additional lighting through ten power LEDs mounted on the chassis but not used for the specific application reported.

Fig. 3. The HW platform block diagram

3.1 Stereo camera calibration

The first procedure always needed to start working with a vision system is to perform the calibration in order to identity both the intrinsic and the extrinsic parameters. The calibration procedure has been performed using the Matlab® camera calibration toolbox. The intrinsic parameters identified include the lens distortion map, the principle points coordinates (C) for the two sensors and the focal lengths (f) in pixel units [Heikkilä & Silvén, 1997], [Zhang, 1999]. For simplicity, the lens distortion function has been assumed to be radial, identified by a sixth order polynomial coefficients (k_i) containing only even exponential terms. As shown by the relation (1), the normalized point (p_n) in image space is required to find the corresponding distorted points (p_d) in the distortion map.

$$p_d = p_n \cdot \left(1 + k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6\right) + C \tag{1}$$

in which

$$p_n = \begin{bmatrix} p_x - C_x \\ \\ p_y - C_y \end{bmatrix}, \quad r = |p_n| \tag{2}$$

Typically, one or two coefficients are enough to compensate for the lens distortion, but in the actual case of the fisheye lenses adopted, all three coefficients have been used. The comparison between the use of two (fourth order distortion model) instead of three (sixth order distortion model) terms in the polynomial map (1), is shown in Fig. 4. The fourth order model is unable to compensate for the strong distortion introduced by the fisheye lens (Fig. 4 - c), which is correctly compensated by the sixth order model (Fig. 4 - d).

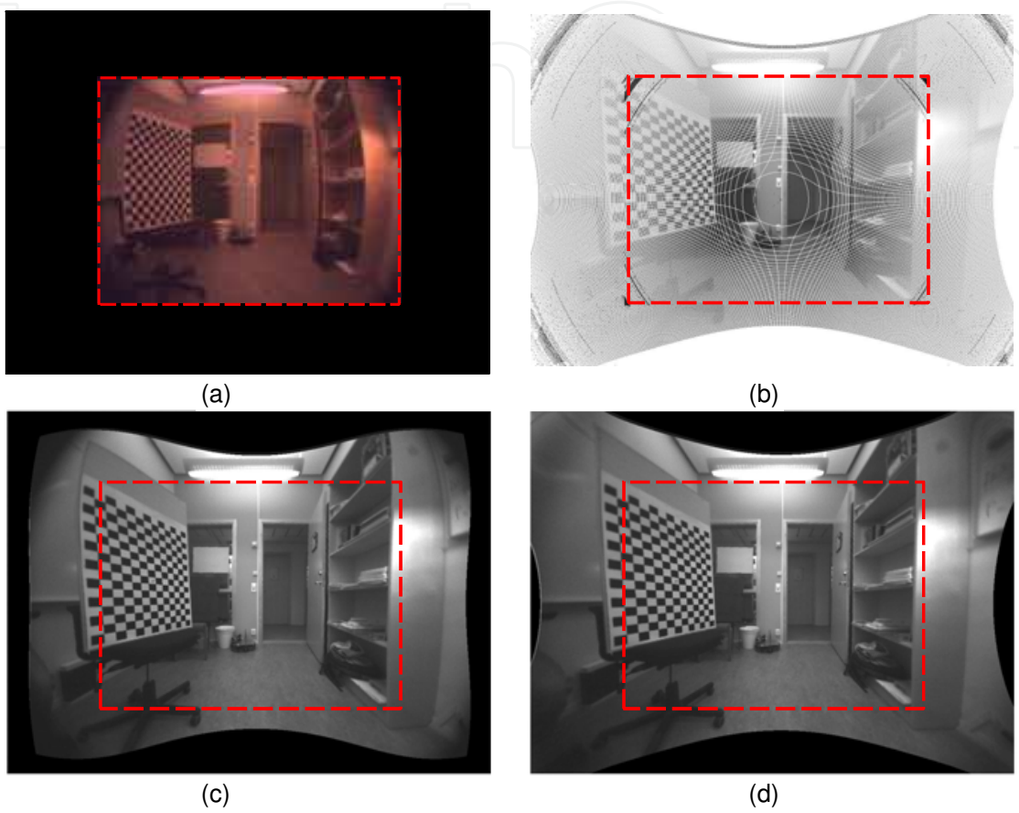


Fig. 4. Original image (a), fourth order distortion compensation (c), sixth order distortion compensation (d). The red squares indicate the original image size. (b) shows the undistorted image without bilinear interpolation.

It is worth to note that the original image size has been expanded by a factor of 1.6 (1024x768) in order to use all the visual information acquired. The principle point has been rescaled according to the new image resolution. The red square in Fig. 4, shows the original image size 640x480.

The undistortion procedure is applied online through an undistortion look up table pre-computed offline using the iterative algorithm described in [Heikkilä & Silvén, 1997] reversing the relation (1). Once the lens distortion has been correctly identified and compensated, the camera system can be used as a standard projective camera, and the pin-hole camera model has been adopted. According to the projective geometry, the 3x4 camera matrix P relates the point p in the image space with the feature F in the 3D space both in homogenous coordinates [Kannala et al., 2009]. Such a matrix is calculated according to the equations (3), where R and T represent the camera pose in terms of rotation and translation with respect to the global reference frame, also known as the extrinsic parameters identified by the calibration procedure.

$$K=\begin{bmatrix} f_x & 0 & C_x \\ 0 & f_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \text{ and } P=K\cdot\begin{bmatrix} R & T \end{bmatrix} \tag{3}$$

$$p=\begin{bmatrix} p_x \\ p_y \\ 1 \end{bmatrix}=\tilde{F}/\tilde{F}_z=\begin{bmatrix} \tilde{F}_x/\tilde{F}_z \\ \tilde{F}_y/\tilde{F}_z \\ 1 \end{bmatrix} \text{ in which } \tilde{F}=\begin{bmatrix} \tilde{F}_x \\ \tilde{F}_y \\ \tilde{F}_z \end{bmatrix}=K\cdot\begin{bmatrix} R & T \end{bmatrix}\cdot\begin{bmatrix} F_x \\ F_y \\ F_z \\ 1 \end{bmatrix}=P\cdot F \tag{4}$$

In the simple case of $R=I$ and $T=[0\ 0\ 0]^T$ the relation (4) yields

$$p=\begin{bmatrix} p_x \\ p_y \\ 1 \end{bmatrix}=\begin{bmatrix} f_x\cdot\dfrac{F_x}{F_z}+C_x \\ f_y\cdot\dfrac{F_y}{F_z}+C_y \\ \\ 1 \end{bmatrix} \tag{5}$$

According to the stereo vision conventions, the translation and rotation matrices R and T represent the position and orientation of the right camera with respect the left one, whereas the global reference frame is placed on the center of the left camera image sensor, giving $R=I$ and $T=[0\ 0\ 0]^T$ as left extrinsic parameters. Extending the relations (3) to the stereo system, the left and right camera matrices can be expressed as $P_R=K_R\cdot\begin{bmatrix} R & T \end{bmatrix}$ and $P_L=K_L\cdot\begin{bmatrix} I & 0 \end{bmatrix}$.

4. Feature extraction

The Harris and Stephens combined corner and edge detection algorithm [Harris & Stephens, 1988] has been implemented in hardware on the FPGA working in real-time. The purpose is to extract the image features in a sequence of images taken by the two cameras for subsequent stereo matching and triangulation. The algorithm is based on a local autocorrelation window and performs very well on natural images. The window traverses the image with small shifts and tests each pixel by comparing it to neighbouring pixels. A Gaussian filter returns the most distinct corners within a projected 5x5 pixels window sliding over the final feature set. Pixels whose strength is above an experimental threshold are chosen as visual features.

To gain real-time speed of the system, the algorithm is designed as a pipeline, so each step executes in parallel. (Three different window generators are used for the derivative, factorization, and comparison masks).

The resulting corner detector algorithm is powerful and produces repeatable features extraction. Fig. 5, shows the block diagram of the feature extractor.

The core of the algorithm is based on the autocorrelation window M that makes use of the horizontal (along the rows: $\partial I/\partial X$) and vertical (along the columns $\partial I/\partial Y$) partial derivatives as shown in Fig. 6.

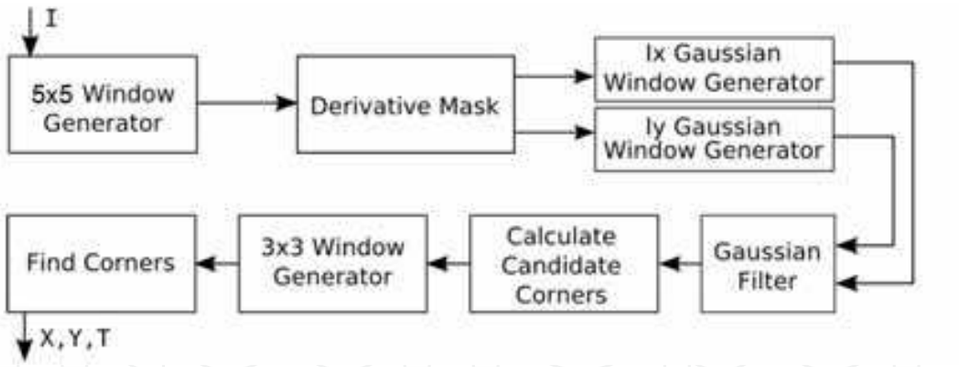


Fig. 5. Stephens-Harris features extractor block diagram

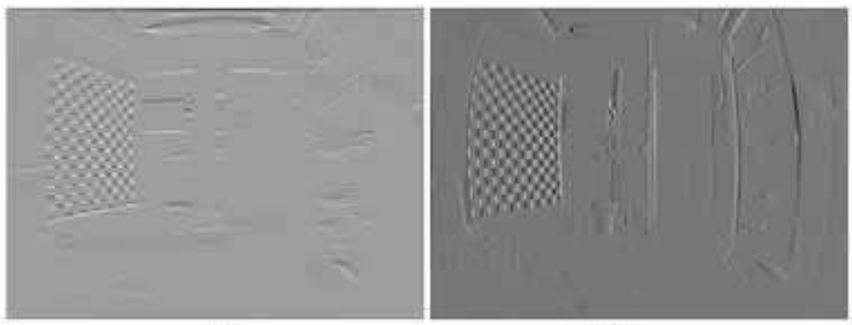


Fig. 6. Image partial derivatives: horizontal image gradient (a), vertical image gradient (b)

From the autocorrelation mask M and its convolution with the Gaussian kernel G (6) two methods for extracting the "*cornerness*" value R against a fixed threshold are universally accepted by the research community: the original method from Harris and Stephens [Harris & Stephens, 1988] (7) and the variation proposed by Noble [Noble 1989] (8) in order to avoid the heuristic choice of the k value (commonly fixed to 0.04 as suggested in [Harris & Stephens, 1988]).

$$M = \begin{bmatrix} \left(\frac{\partial I}{\partial X}\right)^2 & \left(\frac{\partial I}{\partial X}\cdot\frac{\partial I}{\partial Y}\right) \\ \left(\frac{\partial I}{\partial X}\cdot\frac{\partial I}{\partial Y}\right) & \left(\frac{\partial I}{\partial Y}\right)^2 \end{bmatrix} \bullet G \tag{6}$$

$$R = det(M) - k\cdot\left[Tr(M)\right]^2 \tag{7}$$

$$R = \frac{det(M)}{Tr(M)+\varepsilon} \tag{8}$$

As shown if Fig. 7, the choice of the two methods for extracting the "*cornerness*" value are rather equivalent and both effective for the case analyzed in our proposed applications. The main difference is the dynamic threshold that has to be three magnitude orders more in case (7) than (8). This is due to the division that keeps the "*cornerness*" lower.

In Fig. 7 the original Harris is reported in (a) and (b) whereas the Noble case in (c) and (d). On the left side the result of the processing after the autocorrelation M is shown, whereas on the right side, the Harris corners extraction after the thresholding is reported. In the Harris case, a threshold around 10^6 has been applied, whereas 10^3 has been used in the Noble case.

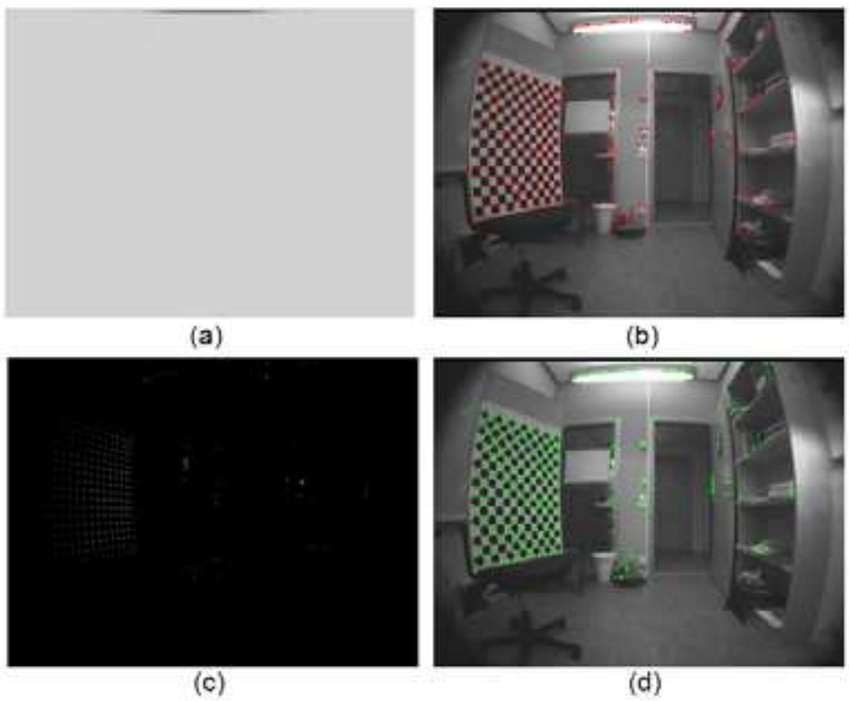


Fig. 7. Comparison between the two methods for corner extraction : Original Harris (a,b), and the Noble variant (c,d).

In our implementation we decided to implement the original method (7) by Harris and Stephens since the division implementation of (8) in the FPGA would have required a lot more resources more.

5. Stereo matching

After the features extraction, the matching of the interest points in the different cameras has to be performed. This phase is essential in stereo and in multiple views based vision, and represents an overhead with respect to a monocular solutions. On the other hand, the matching process acts as a filter removing the most of the noise produced from the feature extraction, since only the strongest features are matched. Although increasing the computational load, the stereo matching increases the process robustness as well.
Two different techniques have been implemented and tested to perform the stereo matching between the left and right images from the stereo rig. Once the features have been extracted from the images, the ICP (Iterative Closest Point) algorithm has been applied to the feature points in order to overlap the two point constellations and find a rigid transformation (rotation and translation) between the images. Since the images are just undistorted but not rectified, a fully rigid transformation (rotation and translation) is needed. The resulting disparity between the two images is considerably reduced, as the example shown in Fig. 8,

so that the correlation based matching on the transformed feature points results in a reduced number of outliers, since the maximum search distance for matching is reduced. A typical reduction in the search distance using this technique is about 70%. Fig. 8 shows sequentially the undistorted stereo images, their overlap from the ICP, and the final features correspondences.

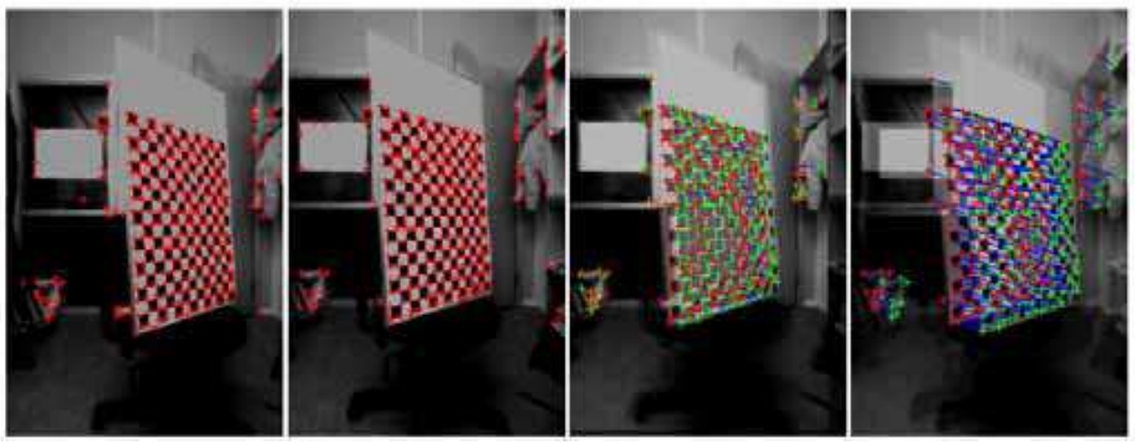


Fig. 8. Stereo matching using the ICP algorithm and correlation

The matching of the feature points is based on the normalized cross correlation (9) computed over an 11x11 window that yields the number of rows R and columns C in (9). About 80% of the corresponding features are correctly matched.

$$Corr = \sum_{r=1}^{R}\sum_{c=1}^{C} \frac{IL(p_{rc}) \cdot IR(p_{rc})}{\sqrt{\sum_{r=1}^{R}\sum_{c=1}^{C} IL(p_{rc})} \cdot \sqrt{\sum_{r=1}^{R}\sum_{c=1}^{C} IR(p_{rc})}} \tag{9}$$

To reduce the computational load, the matching algorithm has been implemented to work directly within the feature space using binary images. The advantage of using this approach is to simplify the cross correlation implementation in the FPGA by reducing the amount of information. The binary images are compared with the XOR bitwise operator instead of the binary multiplication, as shown in (10).

$$Corr = \sum_{r=1}^{R}\sum_{c=1}^{C} \frac{not\left\{XOR\left[IL(p_{rc}), IR(p_{rc})\right]\right\}}{R \cdot C} \tag{10}$$

One example of this technique is shown in Fig. 9, where the ICP is applied to edge reference points and matched with an 11x11 correlation window according to (10). In this case, only 60 % of the corresponding features are correctly matched.

Although the ICP algorithm performs quite robustly, due to its iterative nature, it is time consuming and it is difficult to be parallelized for the implementation into the FPGA.

Another option is to use the epipolar constraint on the undistorted images, using the intrinsic and extrinsic parameters obtained from the calibration. The essential and the fundamental matrices are computed according to (11) and (12) respectively.

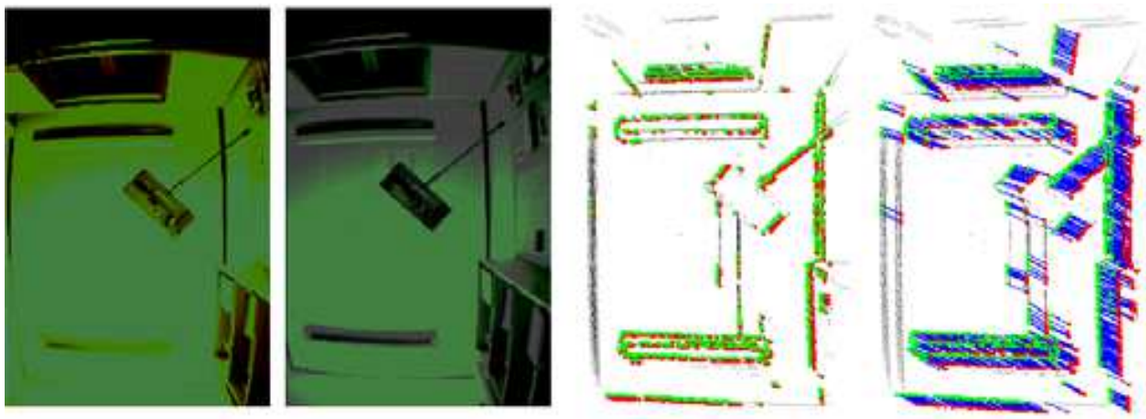


Fig. 9. Stereo matching through ICP and correlation using binary images.

$$E = S(T) \cdot R \tag{11}$$

$$F = K_R^{-T} \cdot E \cdot K_L^{-1} \tag{12}$$

As well known from the multiple view projective geometry theory [Hardley & Zisserman, 2000], for each feature extracted from the left image, the corresponding point in the right image lies on the corresponding epipolar line on the right image whose analytical coefficients are easily extracted from the fundamental matrix *F*. Instead of using the ICP algorithm, a proper search window has to be defined in order to apply the correlation function (10) to the possible corresponding candidates along the epipolar line. The search window is defined to be large enough to cover the maximum disparity at the investigated depth of view. The search window is heuristically defined and strongly depends on the interested depth and the camera vergence. The proposed system has theoretically zero

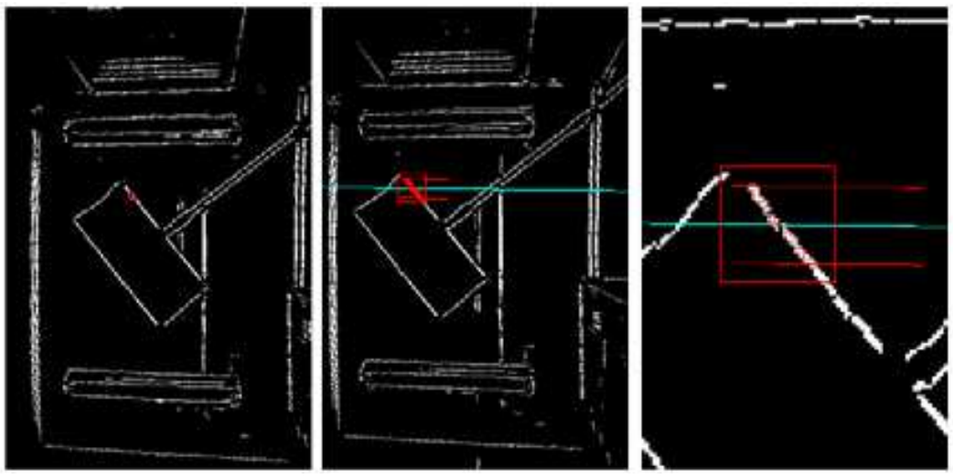


Fig. 10. Stereo matching using the epipolar constraint and the correlation on candidate matches. The search window is represented by the two lines (red) under and above the epipolar line (blue), and the contained candidates within the window are marked in red. Also the correlation window is shown around the correspondent feature indicated by the red square.

vergence due to calibration so that the corresponding feature in the right camera always lies on the left side along the epipolar line with respect to the left feature coordinates (this is not the case of stereo cameras with non zero vergence). In Fig. 10 an example of the described technique is shown. The left feature in the image defines the epipolar line in the right image, as well as the related search window along the epipolar line.

6. Stereo triangulation and depth error modelling

After the corresponding features in the two images are correctly matched, the stereo triangulation can be used to project the interest points in the 3D space. Unfortunately the triangulation procedure is affected by an heteroscedastic error [Matei & Meer, 2006], [Dubbelman & Groen, 2009] (non homogeneous and non isotropic) as shown in Fig.11. An accurate error analysis has been performed in order to provide an uncertainty modelling of the stereo system to the subsequent mapping algorithms that are based on probabilistic estimations. Both 2D and 3D modelling has been investigated.

Knowing the feature projections in the left and right images x_L and x_R, the two dimensional triangulated point P can be found by the well known relations (13), as a function of the baseline b and the focal length f.

$$P_X = \frac{x_L + x_R}{x_L - x_R} \cdot \frac{b}{2} \qquad P_Z = \frac{b \cdot f}{x_L - x_R} \tag{13}$$

$$P_X(s) = \frac{x_L \pm s + x_R \pm s}{x_L \pm s - x_R \mp s} \cdot \frac{b}{2} \qquad P_Z(s) = \frac{b \cdot f}{x_L \pm s - x_R \mp s} \tag{14}$$

A noise error $\pm s$ has been added to the features coordinates in both images, and the resulting noise in the triangulation is represented by a rhomboid whose shape is analytically described by eight points obtained appropriately adding and subtracting the noise s to the nominal image coordinates through (14). The diagonals D and d in Fig.11 represent the corresponding uncertainty in the space reconstruction. The vertical and horizontal displacements H and W in Fig.11 show the heteroscedastic nature of the reconstruction noise since they have different analytical behaviours (non isotropic in the two dimensions) and non linear variations for each point along the two axis (non homogeneous).

$$\begin{aligned} d(s) &= 2\frac{s}{f} \cdot P_Z \qquad D(s) = \sqrt{H^2 + W^2} \\ H(s) &= \frac{4 \cdot b \cdot f \cdot s \cdot {P_Z}^2}{b^2 \cdot f^2 - 4 \cdot s^2 \cdot {P_Z}^2} \qquad W(s) = \left| P_X \cdot \frac{2 \cdot s}{x_L - x_R - 2 \cdot s} \right| + \left| P_X \cdot \frac{2 \cdot s}{x_L - x_R + 2 \cdot s} \right| \end{aligned} \tag{15}$$

It is worth to note that the error along the horizontal axis is the maximum between d and W and coincides with d in all the points that are triangulated between the two cameras (with an horizontal coordinate within the baseline).

To better analyse the heteroscedastic behaviour of the stereo system adopted, the rhomboid descriptive parameters (H,W,D), are presented in Fig. 12 as a function of the reconstructed point P in the plane in front of the cameras for an error of three pixels.

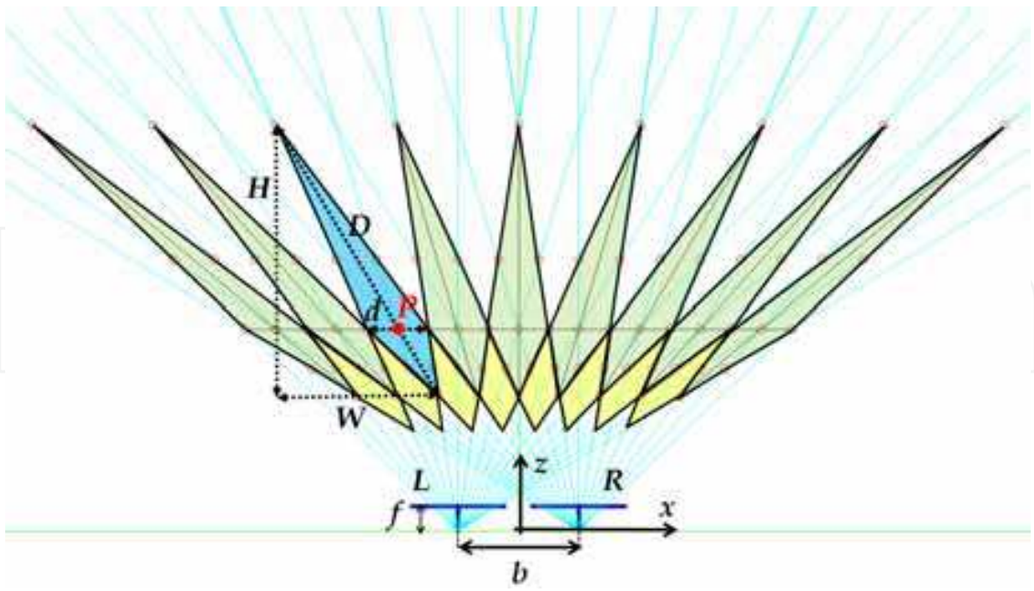


Fig. 11. 2D depth error in stereo triangulation. Two depths of view are reported.

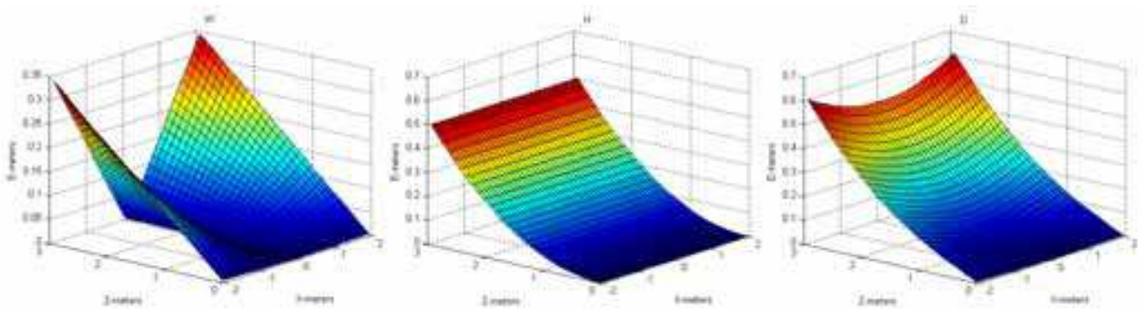


Fig. 12. The descriptive parameters of the rhomboid. From left to right: the horizontal, vertical, and diagonal errors. As expected, only the vertical error remains constant along the horizontal axis while growing non linearly along the vertical axis.

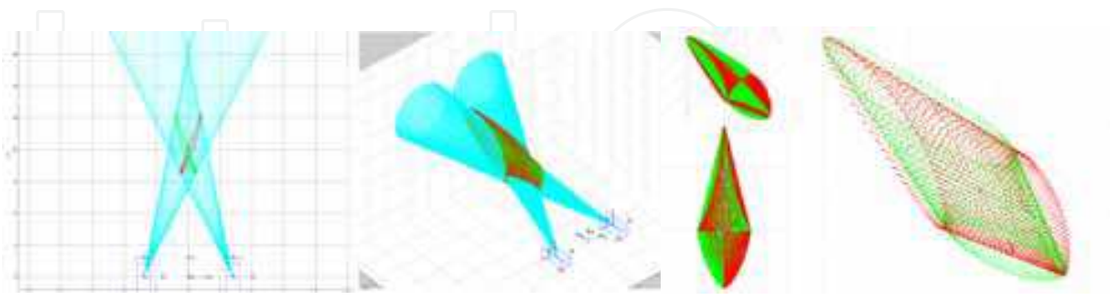


Fig. 13. 3D uncertainty model due to a circular uncertainty in the left and the right images. Matching the feature point in one camera with the circle in the other camera results in the projected ellipse reported inside the 3D intersection region.

Leaving the epipolar plane, the stereo triangulation in 3D space requires a more complex solution in the triangulation procedure since the projective lines could be skew lines in absence of epipolar constraints. Also a more complex 3D error modelling is derived in the

3D space. The feature points affected by a circular noise of certain radius produces two uncertainty circles in the left and in the right images. The corresponding 3D uncertainty is a solid intersection of the two cones obtained projecting the two circles. As direct extension of the two dimensional rhomboid, the solid shape reported in Fig. 13 represents the triangulation uncertainty in 3D space.

The triangulation procedure makes use of a least square solution to minimize reprojection error in both images. The initial hypothesis comes from the extrinsic parameters R and T that relates the two image planes $P_R = R \cdot P_L + T$, that can be rewritten as $P_{ZR} \cdot F_R = R \cdot P_{ZL} \cdot F_L + T$ using the projective transformations for each image plane.

$$F = \begin{bmatrix} F_x & F_y & 1 \end{bmatrix}^T = \begin{bmatrix} \frac{x}{f} & \frac{y}{f} & 1 \end{bmatrix}^T = \begin{bmatrix} \frac{P_x}{P_z} & \frac{P_y}{P_z} & 1 \end{bmatrix}^T \tag{16}$$

Using the matrix formulation the problem can be rewritten.

$$\begin{bmatrix} F_R - R \cdot F_L \end{bmatrix} \cdot \begin{bmatrix} P_{ZR} \\ P_{ZL} \end{bmatrix} = T \tag{17}$$

Posing $A = \begin{bmatrix} F_R - R \cdot F_L \end{bmatrix}$ and solving using the LSM, the 3D point P can be computed both in the left and right reference frames.

$$\begin{bmatrix} P_{ZR} \\ P_{ZL} \end{bmatrix} = \left(A^T \cdot A \right)^{-1} \cdot A^T \cdot T \quad \begin{matrix} P_R = F_R \cdot P_{ZR} \\ P_L = F_L \cdot P_{ZL} \end{matrix} \tag{18}$$

To make a systematic analysis of the triangulation accuracy, analytical relations between the uncertainty in the image space and the related uncertainty in 3D space can be computed through the partial derivatives of the stereo triangulation procedure with respect to the feature points in the two images. Through the jacobian matrix J_{PS} (19) computation, it is easy to find the related 3D uncertainty ΔP under a given uncertainty in X and Y coordinates in both images.

$$J_{PS} = \frac{\partial P}{\partial S} = \begin{bmatrix} \frac{\partial P_X}{\partial L_X} & \frac{\partial P_X}{\partial L_Y} & \frac{\partial P_X}{\partial R_X} & \frac{\partial P_X}{\partial R_Y} \\ \frac{\partial P_Y}{\partial L_X} & \frac{\partial P_Y}{\partial L_Y} & \frac{\partial P_Y}{\partial R_X} & \frac{\partial P_Y}{\partial R_Y} \\ \frac{\partial P_Z}{\partial L_X} & \frac{\partial P_Z}{\partial L_Y} & \frac{\partial P_Z}{\partial R_X} & \frac{\partial P_Z}{\partial R_Y} \end{bmatrix} \qquad \Delta P = \begin{bmatrix} \Delta P_X \\ \Delta P_Y \\ \Delta P_Z \end{bmatrix} = J_{PS} \cdot \begin{bmatrix} \Delta R_X \\ \Delta R_Y \\ \Delta L_X \\ \Delta L_Y \end{bmatrix} \tag{19}$$

In Fig. 14 the 3D distribution of the uncertainty along the long diagonal (equivalent to D in the two dimensional case) is reported, showing the heteroscedastic behaviour.

A known grid pattern, shown in Fig. 15, has been used to measure the triangulation error under the hypothesis of three pixels uncertainty in the image space re-projection. For the stereo system adopted, the 3D reconstruction mostly suffers of uncertainty along the long diagonal (equivalent to D in the two dimensional case) of the 3D rhomboid, that is, along the line connecting the centre of the stereo rig and the landmark observed in 3D.

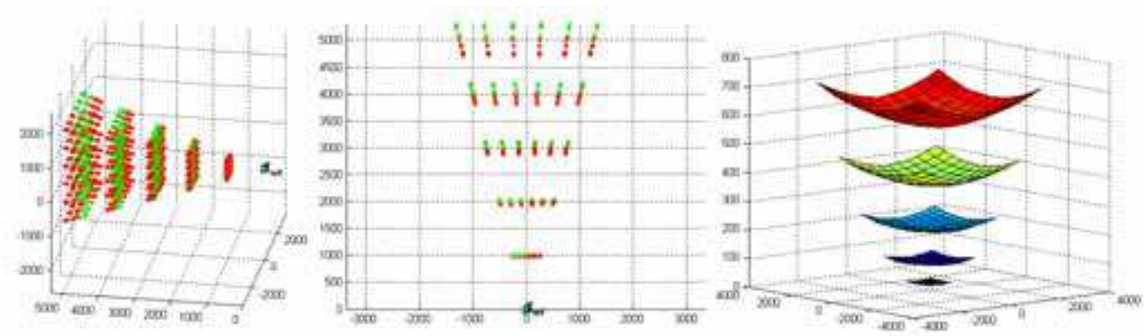


Fig. 14. The 3D uncertainty of the major axis of the ellipsoid related to a grid pattern analyzed at different depths from the cameras.

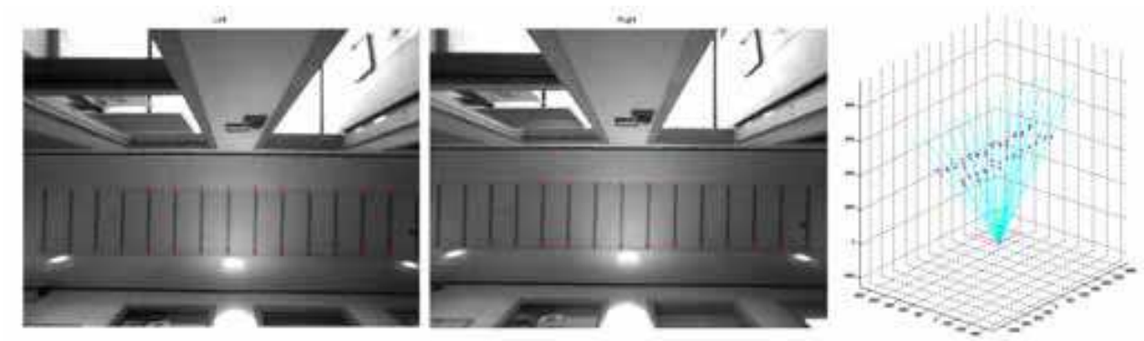


Fig. 15. The reference pattern used to analyse the triangulation error at a 3 m distance from the ceiling.

Extending the reference plane from the ceiling height to arbitrary heights, so that the image projections remain unchanged, the average uncertainty in the three dimensions has been reported in Fig. 16 for distances to the stereo rig from 1 to 30 meters, showing the non linear behaviour as expected. The distribution of the error in the three directions is also presented in the left-most picture for the specific depth of 3 meters.

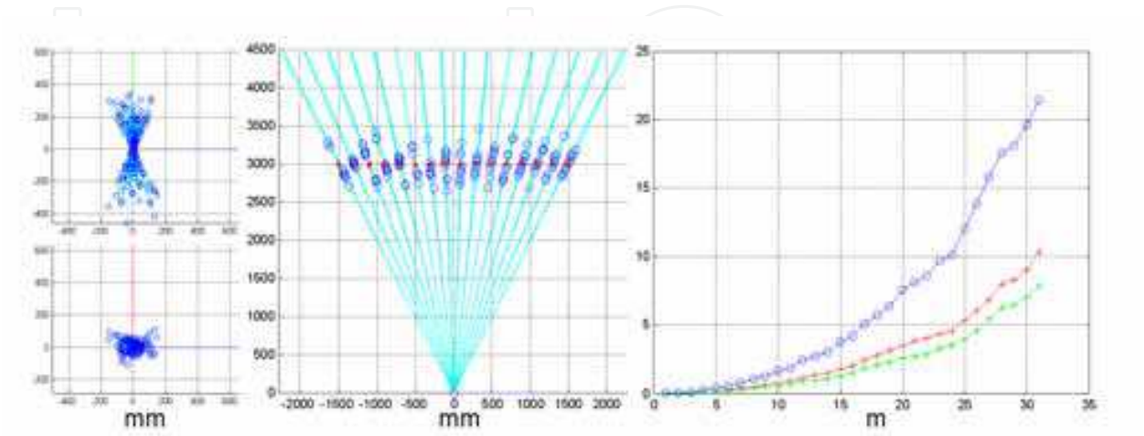


Fig. 16. Distribution of the error along the three dimensions for a fixed depth of view of 3 m; non linear behaviour of the average errors increasing the depth from 1 to 30 m.

7. Visual SLAM

The Simultaneous Localization And Mapping (SLAM) is an acronym often used in robotics to indicate the process through which an automatic controller onboard a vehicle is able to build a map while driving the vehicle in an unknown map or environment and simultaneously localize the robot in the environment.

7.1 Odometry based auto calibration

The SLAM algorithm has been implemented using an Extended Kalman Filter (EKF) based on the visual information coming from the stereo-camera, and using the odometry information coming from the vehicle for simultaneously estimating the camera parameters and the robot landmarks respective positions [Spampinato et al., 2009]. The state variables to be estimated are *3+3N+C*, corresponding to the robot position and orientation (*3* dofs), three dimensional coordinates of *N* landmarks in the environment, and camera parameters *C*, constituting the state vector *x(k)* as shown in (20).

$$x(k) = [X, Y, \vartheta, X_{L1}, Y_{L1}, Z_{L1} \cdots, X_{LN}, Y_{LN}, Z_{LN}, S, \cdots, f]$$

$$u(k) = [V_X, V_Y, V_\vartheta] \tag{20}$$

$$y(k) = \left[F_{R1}, F_{L1}, \cdots, F_{RN}, F_{LN}\right]^T$$

The inputs to the system are the robot velocities for both the position and orientation, whereas the outputs are *4N* feature coordinates on the right and left camera sensors. The model of the system is computed as shown in the relations (21), constituting the *predict phase* of the algorithm.

$$\begin{aligned} x(k+1) &= f(x(k), u(k), k) + v(k) = F(k) \cdot x(k) + G(x, u, k) + v(k) \\ y(k) &= h(x(k), k) + w(k) \end{aligned} \tag{21}$$

The state equations are not linear and generic with respect to the inputs *u(k)* representing the robot generalized velocities. The kinematic model related to the specific vehicle considered is solved a part. The output model is also non linear, and represents the core of the estimator. The state matrix *F(k)* provides the robot position and orientation, computing the corresponding state variables form the input velocities. On the other hand, the landmarks positions and the camera parameters have a zero dynamic behavior.

$$F(k) = \begin{bmatrix} \overset{3\times3}{I} & 0 & 0 \\ 0 & \overset{3N\times3N}{I} & 0 \\ 0 & 0 & \overset{C\times C}{I} \end{bmatrix} \qquad G(x, u, k) = \begin{bmatrix} R(x_3(k)) & 0 \\ 0 & 1 \end{bmatrix} \cdot u(k) \tag{22}$$

The predicted state covariance *P* is a block diagonal matrix, symmetric and positive definite, containing the predicted variances of the state elements.

$$P(k+1) = G_v(k) \cdot P(k) \cdot G_v(k)^T + G_u(k) \cdot v(k) \cdot G_u(k)^T \quad G_v(k) = \left.\frac{\partial f}{dx}\right|_{x=\hat{x}(k)} \quad G_u(k) = \left.\frac{\partial f}{du}\right|_{x=\hat{x}(k)} \tag{23}$$

The system model and the system measurements uncertainties are respectively indicated by the 3x3 diagonal matrix v and the 4x4 diagonal matrix w containing the variances terms. In particular, the model uncertainty are computed basing on the specific kinematics involved, whereas the measurements uncertainty are computed basing on the considerations reported in the previous section regarding the 3D reconstruction accuracy.
During the *update phase* of the EKF, the state variables, and the related covariance matrix P, are updated by the correction from the Kalman gain R and the innovation vector e, as reported by the relations (24).

$$\begin{aligned} \hat{x}(k+1|k+1) &= \hat{x}(k+1|k) + R \cdot e \\ P(k+1|k+1) &= P(k+1|k) - RH(k+1)P(k+1|k) \end{aligned} \tag{24}$$

$$H(k+1) = \left.\frac{\partial h}{dx}\right|_{x=\hat{x}(k+1|k)}$$

The innovation vector represents the difference between the estimated model output h and the real measurements from the stereo camera sensors.

$$\begin{aligned} e &= y(k+1) - h(x(k+1|k), k+1) \\ R &= P(k+1|k)H(k+1)^T S^{-1} \\ S &= H(k+1)P(k+1|k)H(k+1)^T + w(k+1) \end{aligned} \tag{25}$$

The computation of the Kalman gain R, comes from the linearization of the output model around the current state estimation, through the corresponding jacobian matrix H, as presented in (26).

$$\underbrace{\begin{bmatrix} \frac{\partial F_1}{\partial X_R} & \frac{\partial F_1}{\partial Y_R} & \frac{\partial F_1}{\partial \vartheta_R} & \frac{\partial F_1}{\partial L_1} & 0 & 0 & 0 & \frac{\partial F_1}{\partial f} & \cdots & \frac{\partial F_1}{\partial S} \\ \frac{\partial F_2}{\partial X_R} & \frac{\partial F_2}{\partial Y_R} & \frac{\partial F_2}{\partial \vartheta_R} & 0 & \frac{\partial F_2}{\partial L_2} & 0 & 0 & \frac{\partial F_2}{\partial f} & \cdots & \frac{\partial F_2}{\partial S} \\ \vdots & \vdots & \vdots & 0 & 0 & \ddots & 0 & \vdots & \cdots & \vdots \\ \frac{\partial F_N}{\partial X_R} & \frac{\partial F_N}{\partial Y_R} & \frac{\partial F_N}{\partial \vartheta_R} & 0 & 0 & 0 & \frac{\partial F_N}{\partial L_N} & \frac{\partial F_N}{\partial f} & \cdots & \frac{\partial F_N}{\partial S} \end{bmatrix}}_{H:\ 4N \times (3+3N+C)} \cdot \underbrace{\begin{bmatrix} dX \\ dY \\ d\vartheta \\ dL_1 \\ dL_2 \\ \vdots \\ dL_N \\ df \\ \vdots \\ dS \end{bmatrix}}_{dx\ (3+3N+C)\times 1} = \underbrace{\begin{bmatrix} dF_1 \\ dF_2 \\ \vdots \\ dF_N \end{bmatrix}}_{dy\ 4N\times 1} \tag{26}$$

The three groups of parameters to be estimated are quite evident by the structure of the H matrix, where the central part is block diagonal indicating the feature-landmark correspondences.

The camera calibration has been tested on the camera separation estimation using a five LEDs unknown pattern shown in Fig. 17. The camera motion with respect to the landmarks has been performed in a straight path along the X axis.

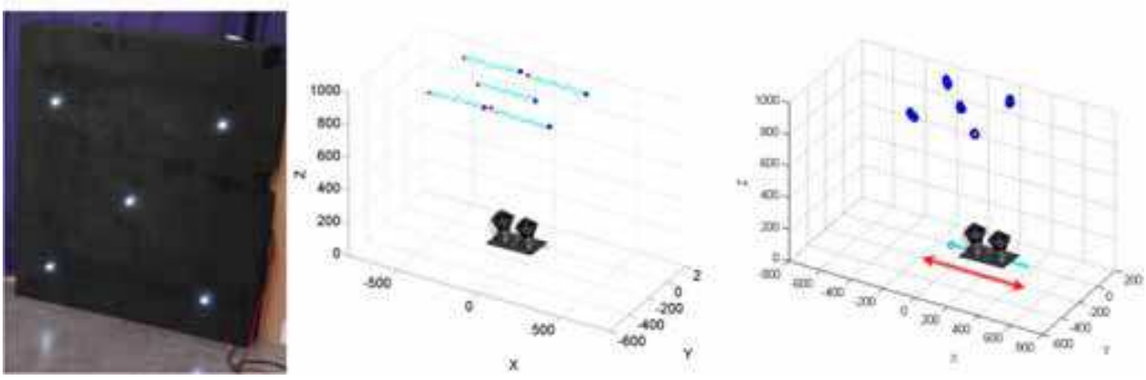


Fig. 17. Landmarks 3D reconstruction with respect to the robot (left) and to the world reference frame (right).

The localization and mapping algorithm has been implemented using the odometry data for the *predict* phase, and the stereo vision feedback for the *update* phase. The state vector is made out of 19 elements, (having one camera parameters $C=1$, and five landmarks $N=5$), representing the three robot DoFs, the 5 three dimensional coordinates of the landmarks, and the camera separation S. Some experimental results are shown in Fig. 17 in which the five landmarks locations are estimated simultaneously with the robot motion back and forth along the X axis, and the camera separation. The position estimation of the central landmark is presented in the upper part of Fig. 18 together with the error with respect to the real three dimensional coordinates. The algorithm errors with respect to the sensor feedback (representing the innovation vector e as described in (25)), are also reported in both the three dimensional space and in the pixels space.

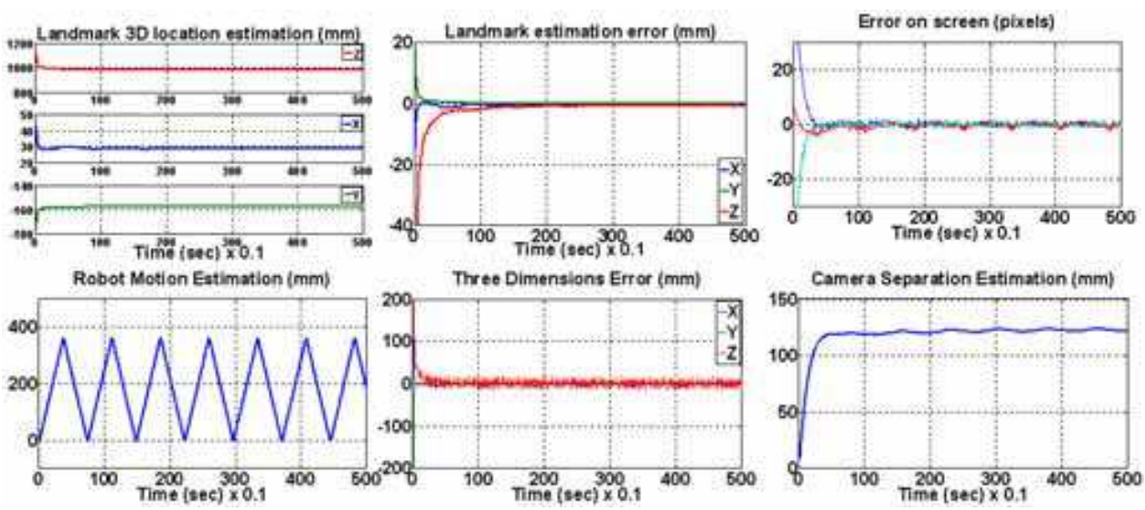


Fig. 18. Experimental results related to the landmarks, robot motion, and camera separation estimation.

7.2 Visual odometry

To keep the whole system simple to use and easy to maintain, more effort has been devoted to avoid to read the odometry data from the vehicle. At the same time, the localization algorithm become more robust to uncertainties that easily arise in the vehicle kinematic model. After the calibration phase, the calibrated stereo rig can be used to estimate the vehicle motion data using solely visual information. The technique, known in literature as *visual odometry* [Nistér et al., 2004], is summarized in Fig. 19 in which the apparent motion of the feature points F in the image space (corresponding to the landmarks P in the 3D space with respect to the vehicle) in two subsequent instants of time, are used to estimate the vehicle motion ΔT and ΔR, in both translation and rotation terms.

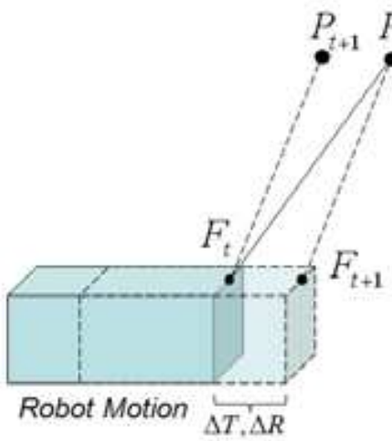


Fig. 19. The visual odometry concept. The vehicle ego-motion is estimated from the apparent motion of the features in the image space.

Back-projecting the features coordinates in the image space to the 3D space using the triangulation described in the previous section, the problem is formalized in estimating the rotation and translation terms that minimize the functional (27).

$$\sum_{i=1}^{n} \left\| P_{t,i} - T - R \cdot P_{t,i+1} \right\|^2 \tag{27}$$

The translation vector is easily computed once the rotation matrix is known, by the distance between the centroids of the two point clouds generated by the triangulated feature points in the subsequent instants of time: $T = \bar{P}_t - R \cdot \bar{P}_{t+1}$, in which the two centroids can be computed as in (28).

$$\bar{P}_{t+1} = \frac{1}{n} \sum_{i=1}^{n} P_{t+1,i} \quad \bar{P}_t = \frac{1}{n} \sum_{i=1}^{n} P_{t,i} \tag{28}$$

The rotation matrix minimizes the functional (29) representing the Frobenius norm of the residual of the landmarks distance with respect to the centroids in the two subsequent instants.

$$\sum_{i=1}^{n} \left\| \bar{P}_{t,i} - R \cdot \bar{P}_{t,i+1} \right\|^2 \tag{29}$$

in which $\overline{P}_{t,i} = P_{t,i} - \overline{P}_t$ and $\overline{P}_{t+1,i} = P_{t+1,i} - \overline{P}_{t+1}$. The rotation term minimizing (29) minimizes also the trace of the matrix $R^T \cdot K$, with $K = \sum_{i=1}^{n} \left(\overline{P}_{t,i}\right)^T \cdot \left(\overline{P}_{t+1,i}\right)$ [Siciliano et al., 2009].

The rotation matrix R is computed through the right and left eigenvector matrices from the SVD of the matrix K, $svd(K) = U \cdot \Sigma \cdot V^T$.

$$R = U \cdot \begin{bmatrix} 1 & 0 \\ 0 & \sigma \end{bmatrix} \cdot V^T \text{ in which } \sigma = \det\left(U \cdot V^T\right). \tag{30}$$

The visual odometry strategy, as described above, is computed within the *predict* phase of the Kalman filter in place of the traditional odometry readings and processing from the vehicle, resulting in a reduced communication overhead, during the motion. An increased robustness to polarized errors, coming from the vehicle kinematic model uncertainties, is also gained.

8. Experimental results

In the current version of the platform, the localization system has been implemented on a standard PC, communicating with the stereo camera through USB. The system has been mounted on three different platforms and tested both within university buildings as well as in industrial sites. The system has been tested at Mälardalen University (MDH), Örebro University (ORU) and in Stora Enso paper mill in Skoghall (Karlstad, Sweden).

The localization in unknown environments and the simultaneous map building solely use visual landmarks (mostly using light sources coming from the lamps in the ceiling), and operate without reading the odometry information from the vehicle.

In the working demonstrator at Mälardalen University, the stereo system has been placed on a wheeled table. The vision system looks upwards, extracts information from the lamps in the ceiling, builds a map of the environment and localizes itself inside the map with a precision within the range of 1-3 cms depending on the height of the ceiling. Two different test cases have been provided for small and large environments as shown, in Fig. 20 and Fig. 21. The system is also able to recover its correct position within the map after a severe disturbance like, for example, a long period of "blind" motion as known as kidnapping.

Fig. 20. Simultaneous localization and map building using only visual information at MDH. The table was moved at about 1m/ s producing the map of the room with 9 landmarks on a surface of about 50 m^2

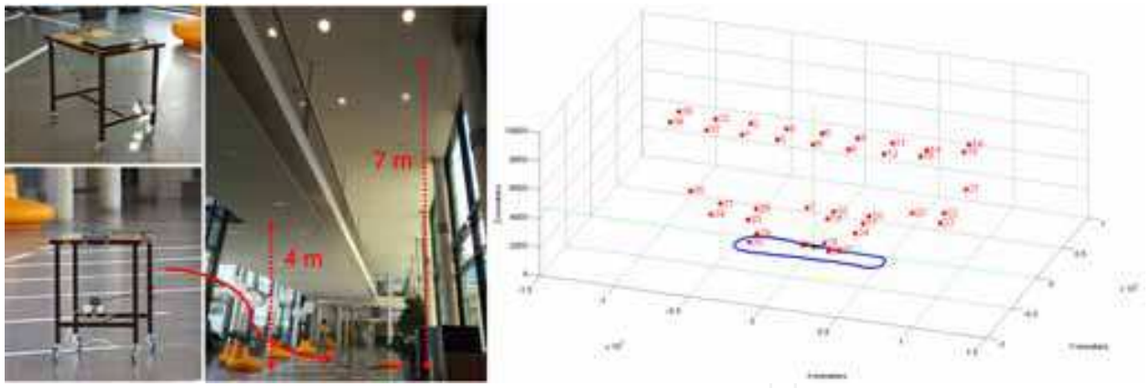


Fig. 21. Simultaneous localization and map building using only visual information at MDH. The table was moved at about 1m/ s producing the map of the university hall with 40 landmarks on a surface of about 600 m². The landmarks are mainly grouped in two layers, at respectively 4 and 7 meters from the cameras.

In the frame of the MALTA project, some experiments have been performed at Örebro university, to test the system when mounted on a small scale version of the industrial vehicle controller used in the project. The robot is equipped with the same navigation system installed by Danaher Motion (industrial partner in the project) in the "official industrial truck", used in the project (the H50 forklift by Linde, also industrial partner in the project).

The system has been tested to verify the vSLAM algorithm to localize and build the map on an unknown environment, and to feed the estimated position to the Danaher Motion system installed in the vehicle as an "epm" (external positioning measurement), and let the robot be controlled by the Danaher system using our localization information, as proof of the reliability of our estimation.

The complete map of two rooms employed a total of 26 landmarks on a surface of about 80 m². The precision of the localization system has been proved marking specific positions in the room and using the map built to verify the correspondence. The precision of the localization was about 1 cm. The three dimensional representation of the robot path and the created map during the experiments is shown in Fig 22. The robot was run for about 10

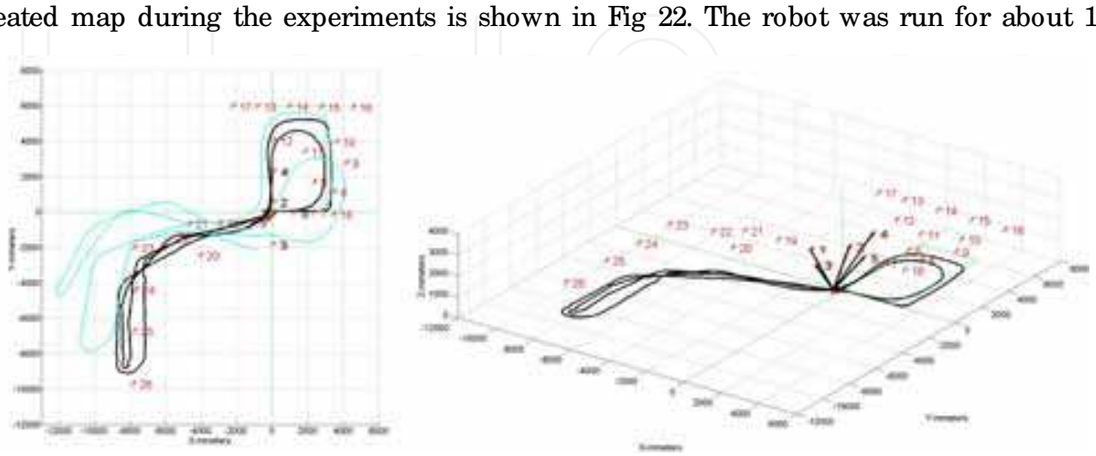


Fig. 22. Three dimensional representations of the robot path and the related map built during the experiments at ORU.

minutes at a speed of 0.3 m/ s. The visual odometry estimated path is also reported to show the drift of the odometry only based estimation with respect to the whole localization algorithm.
Two cubic b-splines trajectories are shown in Fig. 23, while driving the robot using the Danaher Motion navigation system using the proposed position estimation provided as "epm" (external positioning measurement) to the Danaher system. The precision of the localization is within 1 cm.

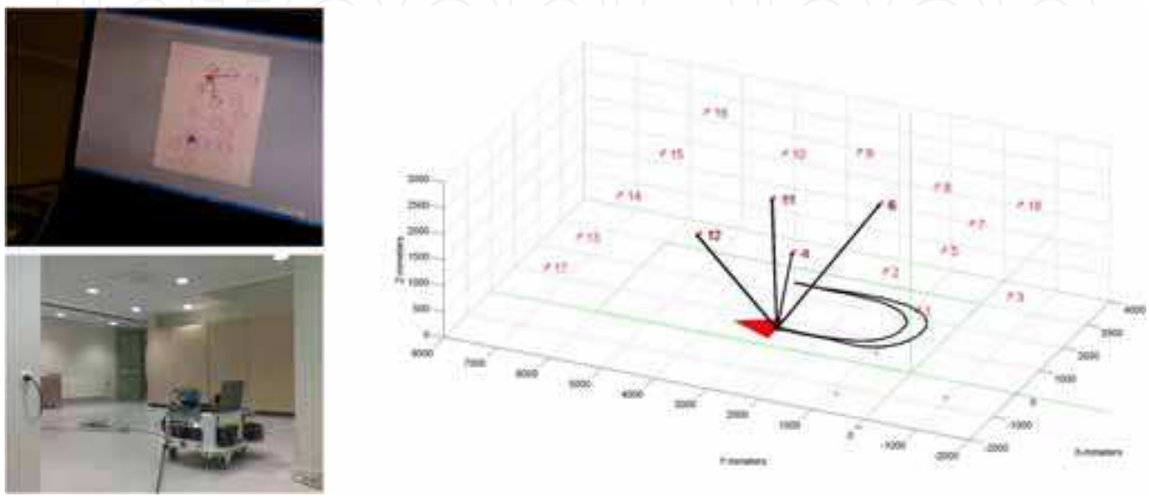


Fig. 23. Three dimensional representations of the robot path and the related map built during two splines based trajectories control executed through the Danaher navigation system and the MDH position estimation provided as "epm".

During the frame of the MALTA project, some tests have been organized in Skoghall, inside the Stora Enso (industrial partner in the project) paper mill to test different localization systems proposed inside the project and also to avoid adding additional infrastructure to the environment.
The vehicle used during the experiments is the H50 forklift provided by Linde Material Handling, and properly modified by Danaher Motion. The tests site, as well as the industrial vehicle used during the demo are shown in Fig. 24. The stereovision navigation system has been placed on top of the vehicle, like shown in the picture, making the system integration extremely easy.
The environmental conditions are completely different from the labs at the universities, and the demo surface was about 2800 m^2. The height of the ceiling, and so the distance of the lamps from the vehicle (used as natural landmarks by our navigation system), is about 20m.
The experiments have been performed, like in the previous cases, estimating the position of the robot and building the map of the environment simultaneously. The estimated position and orientation of the vehicle were provided to the Danaher Motion navigation system as "epm". In Fig. 25 the path estimation is reported while the vehicle was performing a cubic b-spline driving with a speed of 0.5 m/ s. In Fig. 26, a longer path was performed with the purpose to collect as many landmarks as possible and build a more complete map. In this case the complete map employed a total of 14 landmarks on a surface of about 2800 m2 with a precision of about 10 cm.

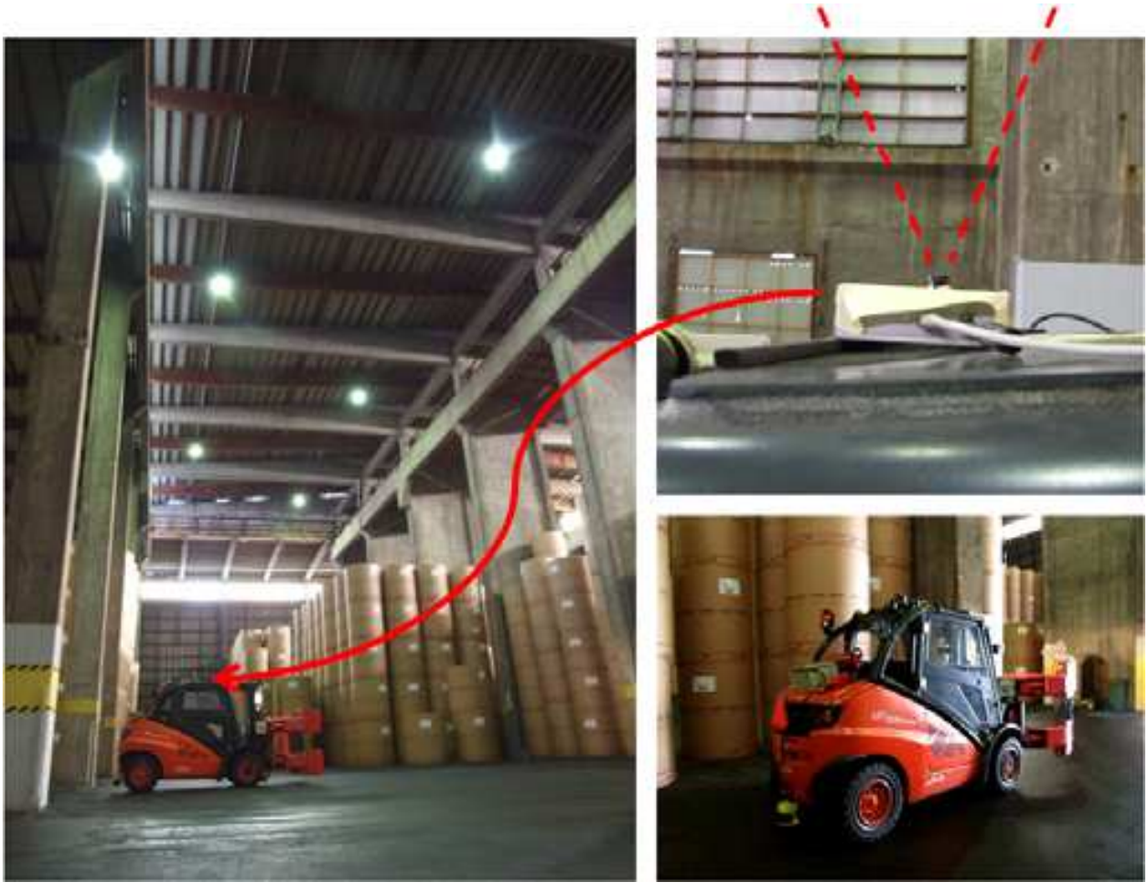


Fig. 24. The demo industrial site in the Stora Enso paper mill in Skoghall (Karlstad, Sweden). The integration of the proposed visual localization system is extremely fast since it is enough to place the stereocamera on top of the vehicle.

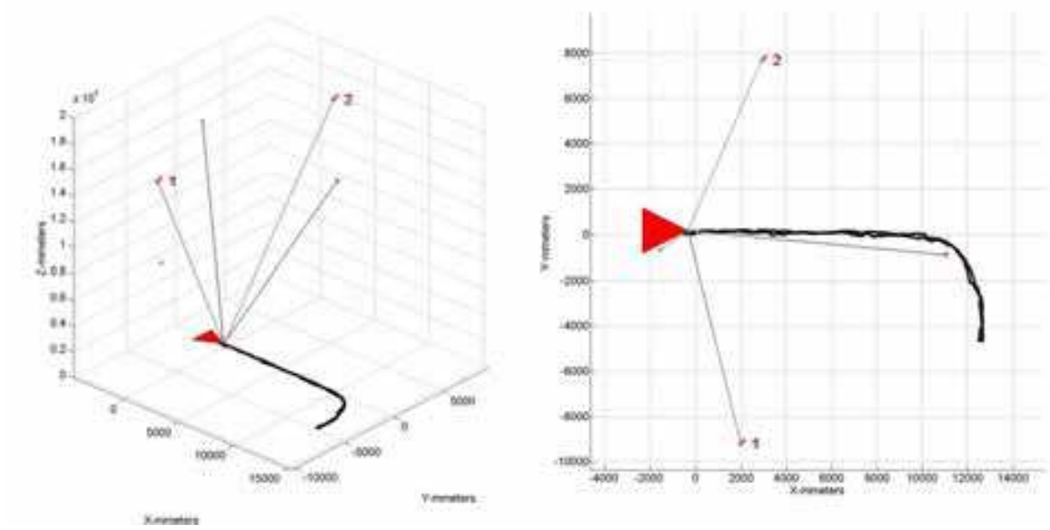


Fig. 25. The planar and three dimensional representation of the vehicle path estimation while performing a b-spline trajectory at 0.5 m/ s inside the Stora Enso industrial site in Skoghall.

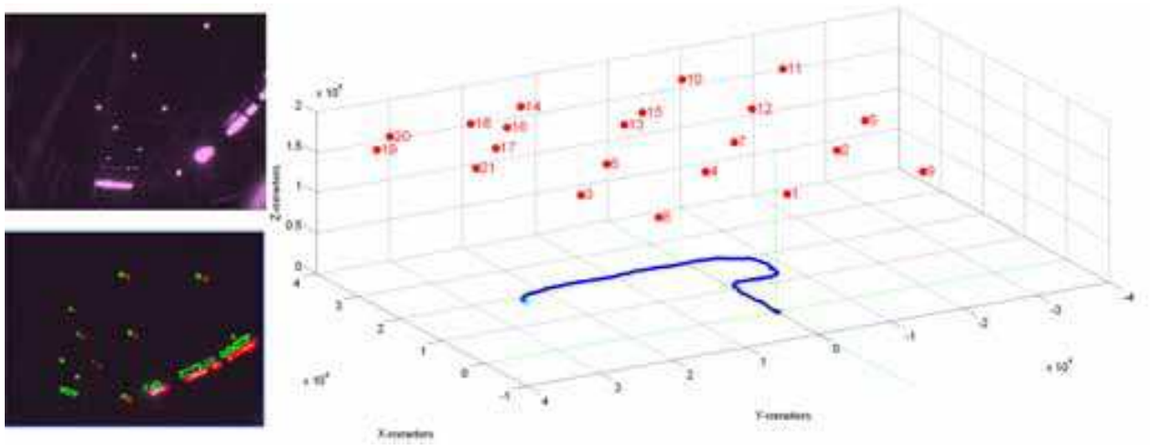


Fig. 26. The three dimensional representation of the vehicle path estimation and map built inside the Stora Enso industrial site in Skoghall. On the left side is shown one screenshot of the feature extraction process.

9. Conclusion

The proposed solution makes use of stereo vision to realize localization and map building of unknown environments without adding any additional infrastructure. The system has been tested in three different environments, two universities and one industrial site. The great advantage for a potential user is the simple installation and integration with the vehicle, since it is enough to place the camera box on the vehicle ad connect via USB to a standard PC to localize the vehicle inside a generated map.

From the industrial point of view, the overall impression is that the precision of the system is good even if the conditions are very different form the lab. The distance from the landmarks is bigger than in the lab, and so the accuracy errors registered. Increasing the speed of the vehicle to 1-1.5 m/ s, the performances of the system severely decrease, resulting in accuracy errors of 30-40 cm from the desired path in the worst cases, that is unacceptable in normal industrial operating conditions.

In order to address the target of autonomous navigation at full speed (30 Km/ h), the core of the vSLAM system needs to be updated to run at a higher frequency (from 3 Hz of the current implementation to 30 Hz), so to speed up also the performances of the "epm" driving mode. Moreover, the USB communication will be substituted with the Ethernet running at 100 Mb/ s, in a closer future. However, in the final version, it is foreseen that the whole system should be implemented in hardware, living the PC as a configuration terminal.

From the algorithmic point of view the next step will update the EKF from 3 DoF to full 6 DoF vehicle position and orientation modeling, in order to compensate for non flat ground and slopes often present in industrial sites.

10. References

Dubbelman, G. & Groen, F.(2009), Bias reduction for stereo based motion estimation with applications to large scale visual odometry. *Proc. Of IEEE Computer Society*

Conference on Computer Vision and Pattern Recognition, pp. 2222–2229, ISBN 978-1-4244-3991-1, Miami, Florida, June, 2009.

Heikkilä J, & Silven O, (1997), A four-step camera calibration procedure with implicit image correction. *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1106-1112, Puerto Rico, San Juan, June, 1997.

Harris, C. & Stephens, M. (1988). A combined corner and edge detection. *In Proceedings of The Fourth Alvey Vision Conference*, pp 147-151, Manchester, UK, 1988.

Hartley, R. & Zisserman, A. (2000). *Multiple View Geometry in Computer Vision*. Cambridge University Press,. ISBN: 0521540518.

Kannala J, Heikkilä J & Brandt S, (2009) Geometric camera calibration. In: *Encyclopedia of Computer Science and Engineering*, Wah BW, Wiley, Hoboken, NJ, 3:1389-1400.

Laugier, C. & Chatila R. (2007), *Autonomous Navigation in Dynamic Environments*. Springer Verlag, ISBN-13 978-3-540-73421-5.

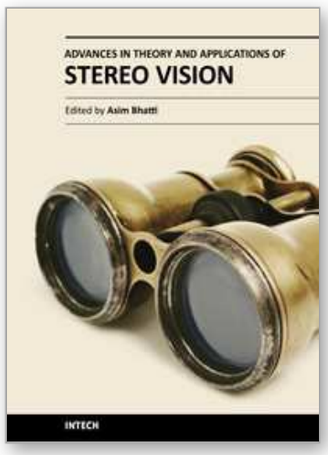
Matei, B. & Meer, P. (2006) Estimation of Nonlinear Errors-in-Variables Models for Computer Vision Applications. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, Vol. 28, No. 10, (October 2006), pp. 1537-1552, ISSN 0162-8828.

Nistér, D.; Naroditsky,O., & Bergen,J. (2004) Visual Odometry. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2004)*, 0-7695-2158-4, Washington DC, USA, June, 2004.

Noble, A. (1989), *Descriptions of Image Surfaces*, PhD thesis, Department of Engineering Science, Oxford University.

Siciliano, B.; Sciavicco, L., Villani,L. & Oriolo,G. (2008) *Robotics: modelling, planning and control. Advanced textbooks in control and signal processing.* Springer, ISBN 1846286417.

Spampinato, G; Lidholm, J, Asplund, L; & Ekstrand,F. (2009) Stereo Vision Based Navigation for Automated Vehicles in Industry. *Proceedings of the 14th IEEE International Conference on emerging Technologies and Factory Automation (ETFA 2009)*, ISBN: 978-1-4244-2728-4, Mallorca, Spain, September, 2009.



Advances in Theory and Applications of Stereo Vision
Edited by Dr Asim Bhatti

ISBN 978-953-307-516-7
Hard cover, 352 pages
Publisher InTech
Published online 08, January, 2011
Published in print edition January, 2011

The book presents a wide range of innovative research ideas and current trends in stereo vision. The topics covered in this book encapsulate research trends from fundamental theoretical aspects of robust stereo correspondence estimation to the establishment of novel and robust algorithms as well as applications in a wide range of disciplines. Particularly interesting theoretical trends presented in this book involve the exploitation of the evolutionary approach, wavelets and multiwavelet theories, Markov random fields and fuzzy sets in addressing the correspondence estimation problem. Novel algorithms utilizing inspiration from biological systems (such as the silicon retina imager and fish eye) and nature (through the exploitation of the refractive index of liquids) make this book an interesting compilation of current research ideas.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Giacomo Spampinato, Jörgen Lidholm, Fredrik Ekstrand, Carl Ahlberg, Lars Asplund and Mikael Ekström (2011). Navigation in a Box: Stereovision for Industry Automation, Advances in Theory and Applications of Stereo Vision, Dr Asim Bhatti (Ed.), ISBN: 978-953-307-516-7, InTech, Available from: http://www.intechopen.com/books/advances-in-theory-and-applications-of-stereo-vision/navigation-in-a-box-stereovision-for-industry-automation

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen