The Seventeenth International Symposium on Artificial Life and Robotics 2012 (AROB 17th '12),
B-Con Plaza, Beppu, Oita, Japan, January 19-21, 2012

Visual IMU in Manhattan-like Enviroments from 2.5D data

Sven Olufs[1] and Markus Vincze[1]

[1] Vienna University of Technology, Automation and Control Institute,
Gusshausstrasse 25-29 / E376, A-1040 Vienna, Austria
(olufs@ieee.org, vincze@acin.tuwien.ac.at)

Abstract: In this paper we presented a novel robust method for a visual IMU in manhattan-like environments i.e. the frequently observed dominance of three mutually orthogonal vanishing directions in man-made environments. Our approach is based on the idea of the separate estimation of the rotational and translational motion based on dense 3D and 2D features. We estimate the Manhattan-like structure by using an MSAC variant that estimates the Manhattan system directly from the 3D data. In contrast to other methods we use only the normal vectors of each voxel rather than estimating it indirectly using plane estimation. In a next step we estimate the translative motion of the robot relative to the Manhattan system using constrained visual odometry. Both rotational and translational motions are fused in a UKF. We show the robustness of our Manhattan-estimation using real world data. In this paper we demonstrate our approach using a Microsoft Kinect, while the approach will work with all kind of 2.5D sensors.

Keywords: Visual IMU, Manhattan-System, Computer Vision

1 INTRODUCTION

The domain of service robotics has become an important and fast growing market in the last decade. While service robotics has become more and more common in the industry domain they are still rare in the home robotics domain. Recent demographic developments in Europe and Japan have shown that there is a need for service robotics in everybody's home due to the elderly society phenomenon. With the home robotics domain we have usually a relative small area (e.g. 100m2), clutter and visually weakly structured environments. To cope with these environments, the use of 2.5D sensors has become quite popular in the last decade. With the recent release of Microsoft's Kinect sensor, the popularity of 2.5D sensors gained a boost. The Kinect sensor is suitable for the task for two reasons: The sensors are cheaper than laser scanners and its offers a depth image at frame rate. The challenge with data from 2.5D data is to cope with noise and uncertainty due to the nature of the sensors. For instance, the quality of 2.5D data from the Kinect depends on the reflection properties of the observed surface and the angle of incidence (assuming Lambert surfaces).

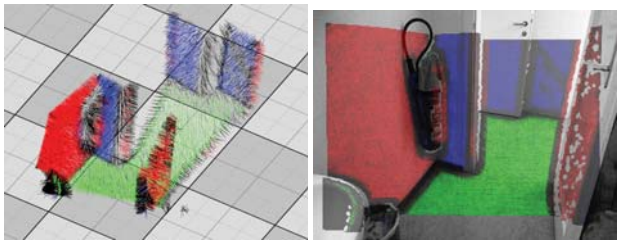
In this paper we propose a novel method to estimate the absolute rotation (namely roll, pitch and yaw) and relative translative motion from 2.5D data by exploiting partial Manhattan-like Geometry of an indoor environment. Manhattan-like structures are the frequently observed dominance of three mutually orthogonal vanishing directions in man-made environments. Many indoor environments can be considered as Manhattan-like since most walls of a room are aligned orthogonally to the ground or quasi Manhattan-like if the walls are not aligned orthogonally to each other. In many cases, furniture is also aligned Manhattan-like to its environment, e.g. a couch or cupboard can be aligned with a wall. Here we emphasize that it is not necessary that the furniture is aligned to all three major axes i.e. even if a table is not aligned to a wall its table surface is usually parallel to the ground.

The novelty of the paper is twofold: First an MSAC variant that directly estimates an Manhattan-system based on normal vectors. In this paper we use it to detect the dominant Manhattan system in the image. The second contribution is a geometric constrained visual odometry in combination with a UKF that exploits kinematic constrains of robot motion. The approach is efficient and robust to noise.

The paper is organized as follows: After a brief discussing on the state of the art, we describe in section 3.1 the proposed approach in two major parts: The first part describes the estimation of the Manhattan system within the 2.5D data. The second part (3.2) presents the constrained visual Odometry. Finally, we give the conclusion and results in section 4.

2 RELATED WORK

Visual odometry is a relative young area in the field of robotics. A poplar approach is the estimation using motion models (top-down). Davison et al. [1] proposed a method where a underlying motion model predicts the position of features to the next frame and updates it with a EKF. This idea was also used by Clipp et al.[2] by using multiple EKF filters or by [3] using more sophist aced features for tracking. Another idea is to use structure from motion techniques



(a) Normal Vectors (b) Image

Figure 1: Estimated Manhattan System of a sample scene. The colors indicate the membership to one of the three major axis.

instead of EKF filter as proposed by Klein et al. [4] or only using spares bundle adjustment Konolige et al. [5] for pose refiement.

The use of the *Manhattan-World* assumption is quite popular in the computer vision literature, for instance, in the use of multi view-reconstruction [6, 7, 8]. Gallup et al. [6] use *Manhattan-World* assumption as prior for plane sweeping i.e. using only orthogonal planes. Furukawa et al. [8] use a similar approach for reconstructing piecewise planar patches and Markov random field formulation for exact planes. Sinha et al. [7] use a similar method, but with a less strict model. Gupta et al. [9] extend the idea by including not a-priory known kinematics on image structure.

3 OUR APPROACH

Our approach consist of three steps: First, we estimate the Manhattan-like structure by using an MSAC variant that estimates the Manhattan system directly from the 3D data. In contrast to other methods we use only the normal vectors of each voxel rather than estimating it indirectly using plane estimation. In a next step we estimate the translative motion of the robot relative to the Manhattan system using constrained visual odometry. Finally we combine both estimates with a unscented Kalman filter to reject implausible motions.

3.1 Manhattan System Estimation

We propose an approach using a variant of the well known Random Sample Consensus (RANSAC) techniques. RANSAC based methods obtain their estimate by randomly selecting coefficients from a given dataset to a known model. The estimation is iterative, in each iteration the number of inlier is counted. After a fixed number of iterations, the model with the most inlier is used as estimate.

The idea is to describe the Manhattan-world as three normal vectors $\vec{N}_1, \vec{N}_2, \vec{N}_3$ one for each axis. We use the normal vectors to express the *orientation* to an axis i.e. the vector is virtually aimed in both directions of the axis. We use a normal vector estimation based on integral images and is a good

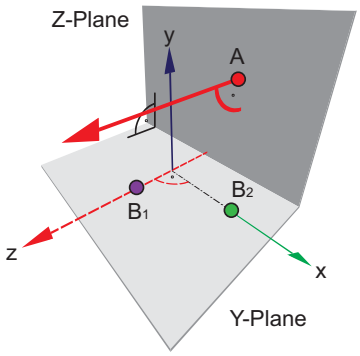


Figure 2: Estimating the Manhattan configuration using three normal vectors and RANSAC methods: A is used as seed for the Manhattan system for the first axis. B_1, B_2 are used to calculate the "roll" of the second axis and the third axis is redundant. Here we assume that A and B_1, B_2 do not share the same orientation plane of the Manhattan system, but that B_1, B_2 does.

trade off of runtime and quality. The normal vector of a voxel counts as an inlier, if the angle to one of the three axis normal vectors is within a certain threshold e.g. 5 degrees. The resulting angle is always between 0-90 degrees, since an axis does not have an orientation like a normal vector. The model is given as follows: Let $\vec{A}, \vec{B}_1, \vec{B}_2 \in V$ randomly selected voxels of the 2.5D grid and $\vec{a}, \vec{b}_1, \vec{b}_2$ its associated normal vectors. The three vectors are calculated with:

$$\begin{aligned} \vec{N}_1 &= \vec{a} \\ \vec{N}_2 &= \vec{B}_2 + \vec{a}((\vec{B}_1 - \vec{B}_2) \cdot \vec{a}) - \vec{B}_1 \\ \vec{N}_3 &= \vec{N}_1 \times \vec{N}_2 \end{aligned}$$

The entire concept is depicted in figure 2: The overall assumption is that A is a point on a Manhattan-like structure for instance on the "Z-Plane". B_1 and B_2 are both on corresponding different Manhattan-like structure for instance the "Y-Plane" or "X-Plane". Since the Manhattan system is redundant to one axis, we only need to calculate two axes e.g. in figure 2 the x axis is redundant. The first vector is given by the normal vector of A itself. The second vector is obtained by shifting the first vector to B_1 and using B_2 as "roll" component. The third vector is the cross product of both previous vectors. This approach generates always a valid Manhattan system using three vectors. Note that we do not check in advance if e.g. B_1 and B_2 are on the same plane, since we have no prior information about planes in the 2.5D data at this step. Such "inplausible configurations" usually generate a Manhattan system with a significant less inliers than a "proper configuration" like in figure 2.

In practice we use MSAC [10] for estimation, an M-

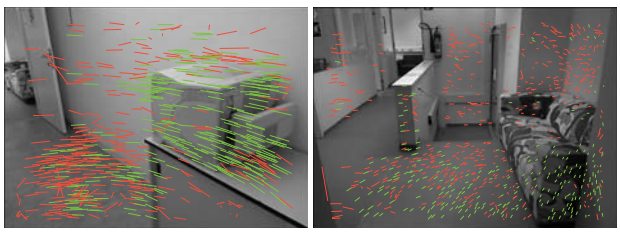
Estimator RANSAC variant: Instead of counting inlier within a specific threshold, we accumulate the error of the model from the original data. The MSAC uses a threshold to specify a maximum error that a voxel belong to the model. Since we assume that we have one dominant Manhattan system we adapt this approach. The idea is to use an additional fixed error if an error exceeds the maximum error. In practice the additional fixed error is significant greater than the measured maximum error according to the threshold. This will raise the probability that the MASC will favor the dominant system rather than a valid random one as long the dominant one is also the largest one in the data. In our setup we use 5 degrees as threshold and 10 times as max constant error. Figure 1 shows an correct estimated Manhattan system.

3.2 Geometric Constratined Visual Odometry

The visual odometry is used to estimate the relative translative motion of the robot to the estimated Manhattan system since the rotation was already obtained on the previous step. We use standard KLT [11] features to track to estimate the motion in the 2.5d data. The tracking is done on the 2D grayscale image using the GPU within $> 1ms$ with Zach et al. [11] implementation. The depth estimation for each point is straight forward, we use the xyz coordinates from the depth data. The motion estimation is done with a one point RANSAC matching with known depth and known correspondence per point. One tracked point is chosen as model i.e. the relative motion of the point in 3D and the previous point while the roation is removed before. As with the standard RANSAC the solution with the most inliers in chosen and used for motion estimation, see figure 3. Only tracked KLT points with valid (non interpolated) depth are used. We use a maximum distance of 10cm for inliers and 50 iterations. The entire runtime for our visual odometry is $> 1ms$.

3.3 Data filtering with UKF

In a first step we track the rational movement and convert it to a angular motion speed for all three angles. The translational part is converted similar to translation speed. In order to cope with either homonymic and non-holonomic robot, we use a motion inertia model, in the fashion of Rao-Blackwellized particle filters [12]. We a central point of mass (like the center of the wheels) as center for all rotations and translational motions. We also assume that the robot is heavy and that a mass lays on top of the center point of mass to prevent that the robot jumpsy up and down in the estimation process. We decompose the six speeds into forces that push the center of mass, plus the gravity (or mass on top) on the center of mass point. Using this sevens force leads to that the UKF filters out implausible motions without expoliting the kinematics to much in detail. We use a straight forward uncended Kalman filter UKF for data fusion, as proposed in various literature e.g. [13]. We don't use an EKF due to it can easly get stuck in local minima, while UKF allows multiple hypothis in the uncended transformation.



(a) Rotative Motion (b) Translative Motion

Figure 3: Visual Odometry using KLT features and RANSAC motion estimation. Outliers are shown in Red, Inliers in Green

4 CONCLUSION AND RESULTS

In this paper we presented a novel robust method for a visual IMU in manhattan-like environments. Our approach is based on the idea of the separate estimation of the rotational and translational motion based on dense 3D and spares 2D features. The proper estimation depends on the amount of visible structure of the Manhattan system within the 2.5D data. As long structure of two axis is visible a robust estimation is possible. As long as Manhattan structure of one axis is clearly visible in the image, its enough that the second axis is partially visible within the data i.e. at least 1 percent of the data. The use of normal vectors is efficient for Manhattan system estimation, but can be computational expensive. The use of integral image style normal vector estimation is a good trade off of runtime and quality and allows to use the entire data set rather than a sub sampled set. Our implementation of the constrained visual odometry is efficient and simple due to only the translative motion part remains.

The overall rotational error is less than 0.2degree, the relative translative error is less than 2mm within a travelling distance of 10m using a Microsoft Kinect. In contrast to other methods we obtain the absolute rotation to the environment instead of the relative rotation like with Visual SLAM. While our work uses the Kinects as sensor, it is also applyable to all kind of 2.5D sensors. We successful used our approach for mapping with a tilting laser scanner using only our visual IMU.

Experiments have shown that many home and office environments contains Manhattan like structure with all three axis. We want to emphasize that the minimal Manhattan system for our approach are two axis, which is very likely in the most indoor environments. That is usually

the ground and a random wall which will be aligned orthogonal in most indoor cases. We figured out that this kind of condition does not hold true in some museums, for instance the Solomon R. Guggenheim Museum in New York.

We also find our Manhattan system estimation every useful in the combination with SLAM in indoor environments. Since our system eliminates to 99.8% the rotation from the data, SLAM has only to deal with the translational mapping error. First experiments with the gmapping package in willow garages Robot Operation System "ROS" have shown improved results with fewer misalignment scans. Due to that gmapping assumes an non-holonomic robot, we had to hack the code to support a psydo-holonomic one.
Our next steps is to relax the Manhattan constrains and to combine our method with SLAM with visual finger prints for place recognition.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community's Seventh Framework Program (FP7/2011-2014) under grant agreement no FP7-ICT-2011-7 288146 (HOBBIT)

REFERENCES

[1] Andrew J. Davison, Ian Reid, Nicholas Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007.

[2] B. Clipp, Lim Jongwoo, J.-M. Frahm, and M. Pollefeys. Parallel, real-time visual slam. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010.

[3] D. Schleicher, L.M. Bergasa, R. Barea, E. Lopez, M. Ocaa, J. Nuevo, and P. Fernandez. Real-time stereo visual slam in large-scale environments based on sift fingerprints. In *IEEE International Symposium on Intelligent Signal Processing*, 2007.

[4] Georg Klein and David Murray. Parallel tracking and mapping for small AR workspaces. In *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)*, 2007.

[5] Kurt Konolige, Giorgio Grisetti, Rainer Kummerle, Wolfram Burgard, and Benson Limketkai and Régis Vincent. Sparse pose adjustment for 2d mapping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010.

[6] D. Gallup, J. Frahm, P. Mordohai, Q. Yang, and M. Pollefeys. Real-time plane-sweeping stereo with multiple sweeping directions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[7] S. Sinha, D. Steedly, and R. Szeliski. Piecewise planar stereo for image-based rendering. In *International Conference on Computer Vision (ICCV)*, 2009.

[8] Y. Furukawa, B. Curless, S. Seitz, and R. Szeliski. Manhattan-world stereo. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[9] A. Gupta, A. A. Efros, and M. Hebert. Blocks world revisited: Image understanding using qualitative geometry and mechanics. In *European Conference on Computer Vision (ECCV)*, 2010.

[10] R. I. Hartley A. and Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[11] C. Zach, D. Gallup, and J.-M. Frahm. Fast gain-adaptive klt tracking on the gpu. In *CVPR Workshop on Visual Computer Vision on GPU's (CVGPU)*, 2008.

[12] Robert Sim and James J. Little. Autonomous vision-based exploration and mapping using hybrid maps and rao-blackwellised particle filters. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2006.

[13] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, Cambridge MA, first edition, 2005.