ON-LINE INCREMENTAL 3D HUMAN BODY RECONSTRUCTION FOR HMI OR AR APPLICATIONS

L. ALMEIDA*†, F. VASCONCELOS†, J. P. BARRETO†, P. MENEZES†, J. DIAS†

†*Institute of Systems and Robotics,*
Department of Electrical and Computer Engineering, University of Coimbra
Polo II, 3030-290 Coimbra, Portugal
E-mail: laa@ipt.pt, {fpv, jpbar, paulo, jorge}@isr.uc.pt

**Department of Informatics Engineering, Institute Polytechnic of Tomar,*
2300 Tomar, Portugal

This research proposes an on-line incremental 3D reconstruction framework that can be used on human machine interaction (HMI) or augmented reality (AR) applications. There is a wide variety of research opportunities including high performance imaging, multi-view video, virtual view synthesis, etc. One fundamental challenge in geometry reconstruction from traditional cameras array is the lack of accuracy in low-texture or repeated pattern region. Our approach explores virtual view synthesis through motion body estimation and hybrid sensors composed by video cameras and a depth camera based on structured-light or time-of-flight. We present a full 3D body reconstruction system that combines visual features and shape-based alignment. The proposed mesh generation algorithm is based on Crust and efficiently adds new vertices to an already existing surface. Modeling is based on meshes computed from dense depth maps in order lower the data to be processed and create a 3D mesh representation that is independent of view-point.

Keywords: Augmented Reality; 3D reconstruction; tele-presence; virtual view synthesis

1. Introduction

This work presents an on-line incremental 3D reconstruction framework that can be used on human machine interaction (HMI) or augmented reality (AR) applications. The project, based on recent low cost depth sensors, intends to create a domestic easy to install 3D acquisition and display system that enables socialization and entertainment. Exploring artificial vision, spatial audio and computers graphics techniques enable us to induce sensations of being physical in the presence of other people useful on several domains like elderly loneness minimization problem, tele-rehabilitation[1],[2] education, socialization, 3DTV, entertainment, etc. Phones and internet chat/audio/video conferencing programs (ex: VOIP, NetMeet-

ing, Skype) are not able to create the remote person presence feeling. Means of communications that enable eye contact, gestures reconnaissance, body language and facial expressions are required.

Augmented reality and particularly tele-immersion[34] can provide the technology means that enables users interact remotely and experience the benefits of a face-to-face meeting. In order to aim an incremental on-line 3D human reconstruction solution useful for shared mixed reality workspace[56],[1] we estimate the 3D world information using 2D image sequences and depth information using a depth camera, e.g. a time of flight camera (ToF) or structured light camera. This hybrid approach addresses the geometry reconstruction challenge from traditional cameras array, that is the lack of accuracy in low-texture or repeated pattern regions. The proposed real-time 3D full reconstruction system combines visual features and shape-based alignment. By identifying features in images for which real-world coordinates can be measured, a correspondence between 3D and 2D is set up. Using those annotated 3D points, between consecutive point clouds, it is possible to estimate the motion transformation through a linear, closed form or iterative method, register them on one same referential and create a global model. Correspondence between consecutive image features in images is performed using SURF method.[7] Virtual view synthesis and modeling is based on 3D mesh from dense depth maps in order lower the data to be processed and to create a 3D mesh representation that is independent of view-point.[8]

The proposed mesh generation algorithm is based on an incremental variant of the Crust algorithm,[9] based on the fact that is possible to efficiently add vertices's to a Delaunay Triangulation. The aim is to continuously generate a realistic body model, transfer the model and reconstruct on a remote common display or virtual environment according each users viewpoint by a tracking process. Figure 1 presents an overview of the algorithm.

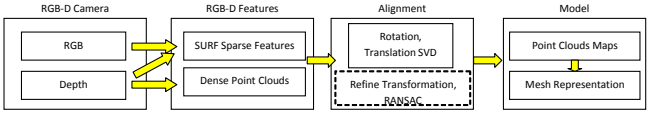


Fig. 1. Algorithm overview. The proposed real-time 3D full reconstruction system combines visual features and shape-based alignment between consecutive point clouds. The model representation is updated incrementally

The reminder of this paper is organized as follows. First some related work, then section 2 describes the suggested methodology and section 3 present some experimental results. Finally, section 4 presents the future work and conclusions.

Background: Virtual view synthesis and modeling are the potential graphic tools to create the eye to eye contact illusion on tele-presence communications.[10] The approach involves surface reconstruction while a basic task for object detection, manipulation and environment modeling. Generally, the object's surface is reconstructed by merging measurements from different views requiring depth data and sensor pose data. When both, pose and depth, are unknown, structure from motion is a solution. Corresponding features in consecutive images are used to estimate the ego-motion of the sensor. Based on this ego-motion information the depth without absolute scale is estimated. Since recent depth cameras also provide RGB data, 2D image processing algorithms are usable. Point feature mapping in RGB images can be improved by the associated depth data obtaining a 3D feature tracking. Most common methods for matching 2D image features are based on the KLT (Kanade-Lucas-Tomasi)[11],[12] SIFT (Scale-Invariant Feature Transform)[13] or SURF (Speeded Up Robust Features)[7] approaches. If only the depth information but no pose is given, i.e. by using a stereo camera or a laser scanner system without inertial sensors, the Iterative Closest Point (ICP) algorithm can be used to register point clouds acquired from different perspectives[14].[15] Finally, if pose and depth are known, the registration procedure is dispensable and the data can simply be merged. Calculating changes in the 3D pose based on these methods have been performed[16][17][18].[19] Our work intends to perform a real-time incremental body modeling.

2. Methodology

We propose a real-time full 3D reconstruction system that combines visual features and shape-based alignment using Xbox Kinect device. Alignment between successive frames is computed by jointly optimizing over both appearance and shape matching.

Registration: Considerer the motion of a rigid body in front of a scanner and the estimation of the rigid transformation (rotation and translation). Assuming that there exist two corresponding 3D point sets $\{\mathbf{x_i^t}\}$ and $\{\mathbf{x_i^{t+1}}\}$, $i = 1..N$, from consecutive t and $t+1$ frames, such that they are related by eq. 1:

$$\mathbf{x_i^{t+1}} = \mathbf{R}\mathbf{x_i^t} + \mathbf{T} + \mathbf{V_i} \quad (1) \qquad \varepsilon^2 = \sum_{i=1}^{N} \left\| \mathbf{x_i^{t+1}} - \mathbf{R}\mathbf{x_i^t} - \mathbf{T} \right\|^2 \quad (2)$$

where $\mathbf{R}$ is a standard 3x3 orthonormal rotation matrix, $\mathbf{T}$ is a 3-D translation vector and $\mathbf{V_i}$ a noise vector. Solving for the optimal transformation $[\mathbf{R}, \mathbf{T}]$ that maps the set $\{\mathbf{x_i^t}\}$ onto $\{\mathbf{x_i^{t+1}}\}$ typically requires minimizing a least squares error criterion given by eq. 2. The minimization of eq. 2 can be based on the singular

value decomposition (SVD) of a matrix.[20] Calculating rotation: as a consequence of the least-squares solution to eq. 2, the point sets $\{\mathbf{x_i^t}\}$ and $\{\mathbf{x_i^{t+1}}\}$ should have the same centroid. Using this constraint a new equation can be generated. By defining:

$$\overline{\mathbf{x_i^t}} = \frac{1}{N}\sum_{i=0}^{n}\mathbf{x_i^t}, \quad \overline{\mathbf{x_i^{t+1}}} = \frac{1}{N}\sum_{i=0}^{n}\mathbf{x_i^{t+1}}, \quad \mathbf{x_{ci}^t} = \mathbf{x_i^t} - \overline{\mathbf{x_i^t}}, \quad \mathbf{x_{ci}^{t+1}} = \mathbf{x_i^{t+1}} - \overline{\mathbf{x_i^{t+1}}} \tag{3}$$

Minimizing this equation is equivalent to maximizing $Trace(\mathbf{R}\,\mathbf{H})$, where $\mathbf{H}$ is a 3x3 correlation matrix defined by $\mathbf{H} = \mathbf{x_{ci}^{t+1}}(\mathbf{x_{ci}^t})^{\mathbf{T}}$. If the singular value decomposition of $\mathbf{H}$ is given by $\mathbf{H=UDV}^T$ then the optimal rotation matrix, $\mathbf{R}$, that maximizes the desired trace is $\mathbf{R=\ U}$ diag(1; 1; det($\mathbf{UV}^T$)) $\mathbf{V}^T$,[20] $\mathbf{R} = \mathbf{UV^T}$.
The optimal translation that aligns the centroid of the sets is $\mathbf{T} = \overline{\mathbf{x_i^{t+1}}} - \mathbf{R}\overline{\mathbf{x_i^t}}$.
Model Mapping: Suppose that the mapping from the world coordinates to one of the scans of the sequence, $^{\mathbf{0}}\mathbf{H_w}$. For any consecutive pair of scans (t, t+1) using tracked points it is possible to measure rotation and translation, $^{\mathbf{t+1}}\mathbf{H_t}$, $\mathbf{H} = [\mathbf{R}, \mathbf{T}]$ (single homogeneous matrix 4x4) and compute Eq. 4:

$$^{\mathbf{i}}\mathbf{H_0} = {}^{\mathbf{i}}\mathbf{H_{i-1}}{}^{\mathbf{i-1}}\mathbf{H_{i-2}}\ldots..{}^{\mathbf{1}}\mathbf{H_0} \qquad \textit{and} \qquad {}^{\mathbf{i}}\mathbf{H_w} = {}^{\mathbf{i}}\mathbf{H_0}{}^{\mathbf{0}}\mathbf{H_w} \tag{4}$$

To update the reconstructed model, each acquired 3D point set is transformed to the world coordinate system using $^{\mathbf{i}}\mathbf{H_w}$. This alignment step adds a new scan to the dense 3D model.*Virtual View Synthesis:* Once having the model, new perspectives views can be presented at each time instant depending on the viewers pose in front of the display (using a head/body tracking module).
Mesh Generation: In order to obtain a surface mesh from the registered 3D point clouds, we use an incremental adaptation of the Crust algorithm.[9] For a given input set of points X, the Crust algorithm determines a set of poles P that lie on the medial axis of the surface. The Delaunay triangulation of $X \cup P$ is then computed, and finally the surface mesh is then obtained by extracting the set of simplices whose vertices belong to X. Given that it is possible to efficiently add vertices to a Delaunay Triangulation,[21] an incremental version of the Crust Algorithm is easy to implement. If the Crust Algorithm was already performed on a set of points X_1, a set of poles P_1 and the Delaunay triangulation of $X_1 \cup P_1$ are available as well. To add a new set of sample points X_2 to the surface mesh, the following steps are performed:

(1) P_2 = poles of X_2
(2) Add $P_2 \cup X_2$ as new vertices of the Delaunay triangulation
(3) Extract triangles whose vertices belong to $X_1 \cup X_2$

This procedure can be repeated for an arbitrary number of point sets X_i. However, in order to avoid an indefinite growth in the number of mesh vertices, points whose

euclidean distance to the closest mesh vertex is lower than a given threshold are deleted from the input cloud before performing the incremental Crust step.
Algorithm:The global model reconstruction algorithm can be described as follow 2.1:

Algorithm 2.1 Model reconstruction algorithm

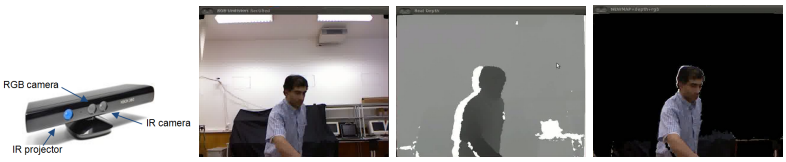
$R_g \leftarrow R_{init}$; $t_g \leftarrow t_{init}$
$f_1 \leftarrow undistort(adquire\_rgb\_image())$, $f_{1d} \leftarrow undistort(adquire\_depth\_image())$
$f_{1xyz} \leftarrow convert\_depth\_image\_to\_xyz\_data(f_{1d})$, $f_{1r} \leftarrow map\_rgbcolor\_to\_depth\_image(f_{1xyz}, f_1)$
for (;;) **do**
  $f_2 \leftarrow undistort(adquire\_rgb\_image())$, $f_{2d} \leftarrow undistort(adquire\_depth\_image())$
  $f_{2xyz} \leftarrow convert\_depth\_image\_to\_xyz\_data(f_{2d})$
  $f_{2r} \leftarrow map\_rgbcolor\_to\_depth\_image(f_{2xyz}, f_2)$

  $(surf_1, surf_2) \leftarrow detect\_SURF\_features(f_{1r}, f_{2r})$
  $matches2D \leftarrow SURF\_match(surf_1, surf_2)$
  $matches3D \leftarrow correspond2D3D(matches3D)$
  $(R,t) \leftarrow motion\_estimator(matches3D)$
  $(R_g, t_g) \leftarrow update\_global\_transformation(R,t)$
  $f_{1r} \leftarrow f_{2r}$; $f_{1xyz} \leftarrow f_{2xyz}$ {update_past_data}
  $model \leftarrow project\_points\_to\_world\_coordinates(f_{2xyz}, R_g, t_g)$
  $mesh\_modelgeneration$
end for

3. Implementation and Results

Novel depth sensors like PrimeSense camera or Xbox Kinect can capture video images along with per-pixel depth information and to experimentally test the algorithm we register several 3D point clouds in order to create person model while he is rotating in front of Kinect device. *Calibrations:* The Kinect device combines a regular RGB camera and a 3D scanner, consisting of an infrared (IR) projector and an IR camera as shown in figure 2a) and the calibration is unavoidable. The aim is to undistort the RGB and IR images and map depth pixels with color pixels (see figure 2). Kinect maximal range raw depth is 2^{11}, and it relates to metric depth through a linear approximation $d_m(x_{ir}, y_{ir}) = f(rawdepth(x_{ir}, y_{ir}))$. From the metric depth, the 3D metric position $(X_{ir}, Y_{ir}, Z_{ir})^T$ of the pixel, with the respect to the IR camera, can be computed using the following equation: $(X_{ir}, Y_{ir}, Z_{ir})^T = (\frac{(x_{ir}-c_{xir})*d_m(x_{ir},y_{ir})}{f_{xir}}, \frac{(y_{ir}-c_{yir})*d_m(x_{ir},y_{ir})}{f_{yir}}, d_m(x_{ir}, y_{ir}))^T$, where x_{ir}, y_{ir} are the coordinates of the depth pixel in image, f_{xir},f_{yir} the IR camera focal length (pixel size units), c_{xir}, c_{yir} the coordinates of the image center of IR camera, and d_m is depth in meters. The IR and RGB cameras are separated by a small baseline, it is possible to determine the 6 DOF transform between them. Knowing the rotation **R** and translation **T** between the RGB and IR camera, we can then re-project each 3D point on the color image and get its color. The

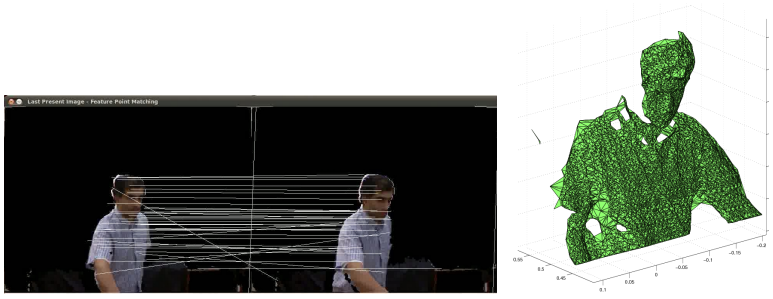
mapping between color image and depth image can be expressed by following equations: $(X_{rgb}, Y_{rgb}, Z_{rgb})^T = \mathbf{R}(X_{ir}, Y_{ir}, Z_{ir})^T + \mathbf{T}$ and ($x_{rgb} = \frac{(X_{rgb} * f_{xrgb})}{Z_{rgb}} + c_{xrgb}$, $y_{rgb} = \frac{(Y_{rgb} * f_{yrgb})}{Z_{rgb}} + c_{yrgb}$), where x_{rgb}, y_{rgb} are the coordinates of the rgb pixel in image, f_{xrgb}, f_{yrgb} the RGB camera focal length, c_{xrgb}, c_{yrgb} the image center, and d_m is depth in meters.



a) *b)* *c)* *d)*

Fig. 2. a) Kinect sensor b) undistorted RGB image c) undistorted depth Image, the body black pixels have unknown depth value, due occlusions or reflective surface material d) Map between undistorted RGB image and depth image.

On figure 3a we present an example of correspondence between consecutive image features in using SURF method (white lines indicate correspondent point). Some matches are undesirable and are related with background static areas. The contribution of erroneous matches is minimized by the number of good matches while using the described minimization method to obtain the transformation. Although SIFT descriptor present better accuracy, we have choose SURF method in order to achieve the real-time characteristic.



a) *b)*

Fig. 3. a) SURF features matched on consecutive time frames b) Mesh model with 27864 vertices and 31810 faces

An example of mesh generation using the proposed incremental adaptation of Crust algorithm is provided on figure 3b with 27864 vertices and 31810 faces.

Figure 4 depicts a sequence of scans that creates a 3D person model. They result from several 3D point clouds fused after applying successive 3D rigid body transformations. Typically the system has a performance of 2 HZ. The time consuming

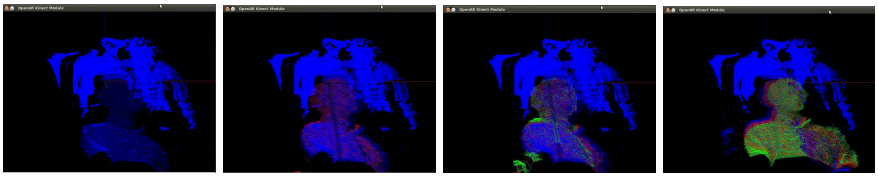


Fig. 4. 3D Model, sequence of point clouds being registered on the same referential, each color represent time sequential scan

stage is related with the surf feature extraction and it takes an average of 300 ms. It depends on the number of detected good feature of the image, although we expect to speed up significantly this step by making use of GPU. The involved number of points also influences the transformation time calculus. On table 1 we present some typically time measure involving some algorithm steps.

Algorithm Steps	**(ms)**
Acquisition	1.55
Undistort Images	10.61
DepthRGB Map and last frame update	36.13
SURF feature extraction	314.853
Matching and transformation calculus	78.0282
Alignment, display and interaction	30.377
Total	471.56 (f=2.12 Hz)

4. Conclusions

There is still a potential for algorithm speedup involving code optimization, GPU CUDA programming and stereo display graphics. Our approach explores virtual view synthesis through motion body estimation and hybrid sensors composed by video cameras and a low cost depth camera based on structured-light, improving the reconstruction accuracy in low-texture or repeated pattern region. We present a full 3D body reconstruction system that combines visual features and shape-based alignment. Modeling is based on meshes computed from dense depth maps in order lower the data to be processed and create a 3D mesh representation that is independent of view-point. This work presents an on-line incremental 3D reconstruction framework that can be used on low cost augmented reality (AR) or HMI applications to enable socialization and entertainment.

References

1. G. Kurillo, T. Koritnik, T. Bajd and R. Bajcsy, *Stud Health Technol Inform* **163**, 290 (2011).
2. A. A. Rizzo and G. J. Kim, *Presence* **14**, 119 (2005).
3. S.-H. Jung and R. Bajcsy, *Journal of Multimedia* **1**, 9 (2006).
4. R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier and B. MacIntyre, *IEEE Comput. Graph. Appl.* **21**, 34(November 2001).
5. G. Kurillo, R. Vasudevan, E. Lobaton and R. Bajcsy, A framework for collaborative real-time 3d teleimmersion in a geographically distributed environment, in *Multimedia, 2008. ISM 2008. Tenth IEEE International Symposium on*, dec. 2008.
6. B. Petit, J.-D. Lesage, J.-S. Franco, E. Boyer and B. Raffin, Grimage: 3d modeling for remote collaboration and telepresence, in *ACM Symposium on Virtual Reality Software and Technology*, October 2008.
7. H. Bay, T. Tuytelaars and L. V. Gool, Surf: Speeded up robust features, in *In ECCV*, 2006.
8. J. Carranza, C. Theobalt, M. A. Magnor and H.-P. Seidel, *ACM Trans. Graph.* **22**, 569(July 2003).
9. N. Amenta, M. Bern and M. Kamvysselis, A new voronoi-based surface reconstruction algorithm, in *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '98 (ACM, New York, NY, USA, 1998).
10. F. Isgro, E. Trucco, P. Kauff and O. Schreer, *Circuits and Systems for Video Technology, IEEE Transactions on* **14**, 288 (march 2004).
11. J. Shi and C. Tomasi, Good features to track, in *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on*, jun 1994.
12. B. D. Lucas and T. Kanade, An iterative image registration technique with an application to stereo vision, in *Proceedings of the 7th international joint conference on Artificial intelligence - Volume 2*, (Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1981).
13. D. G. Lowe, *Int. J. Comput. Vision* **60**, 91(November 2004).
14. P. J. Besl and N. D. McKay, *IEEE Trans. Pattern Anal. Mach. Intell.* **14**, 239(February 1992).
15. J. Lobo, L. Almeida, J. Alves and J. Dias, Registration and segmentation for 3d map building: A solution based on stereo vision and inertial sensors, in *ICRA*, (IEEE, 2003).
16. P. Henry, M. Krainin, E. Herbst, X. Ren and D. Fox, RGB-D Mapping: Using Depth Cameras for Dense 3D Modeling of Indoor Environments (2010).
17. L. G. B. Mirisola, J. L. 0002 and J. Dias, 3d map registration using vision/laser and inertial sensing, in *EMCR*, 2007.
18. A. Akbarzadeh, J.-M. Frahm, P. Mordohai, B. Clipp, C. Engels, D. Gallup, P. Merrell, M. Phelps, S. N. Sinha, B. Talton, L. W. 0002, Q. Yang, H. Stewénius, R. Yang, G. Welch, H. Towles, D. Nistér and M. Pollefeys, Towards urban 3d reconstruction from video, in *3DPVT*, (IEEE Computer Society, 2006).
19. P. Menezes, F. Lerasle and J. Dias, *IVC* **29**, 382(May 2011).
20. K. S. Arun, T. S. Huang and S. D. Blostein, *IEEE Trans. Pattern Anal. Mach. Intell.* **9**, 698(September 1987).
21. L. Guibas, D. Knuth and M. Sharir, *Algorithmica* **7**, 381 (1992), 10.1007/BF01758770.