ARTICLE IN PRESS

Robotics and Autonomous Systems ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Incremental texture mapping for autonomous driving

Miguel Oliveira [a,b,*], Vitor Santos [b], Angel D. Sappa [c,d], Paulo Dias [b], A. Paulo Moreira [a,e]

[a] INESC TEC - INESC Technology and Science, R. Dr. Roberto Frias s/n, 4200-465 Porto, Portugal

[b] IEETA - Institute of Electronics and Informatics Engineering of Aveiro, Universidade de Aveiro, Campus Universitário de Santiago, 3810-193 Aveiro, Portugal

[c] Escuela Superior Politécnica del Litoral, ESPOL, Facultad de Ingeniería en Electricidad y Computación, CIDIS, Campus Gustavo Galindo, Km 30.5 Vía Perimetral, P.O. Box 09-01-5863, Guayaquil, Ecuador

[d] Computer Vision Center, Campus UAB, 08193 Bellaterra, Barcelona, Spain

[e] FEUP - Faculty of Engineering, University of Porto, R. Dr. Roberto Frias s/n, 4200-465 Porto, Portugal

HIGHLIGHTS

- Texture mapping of Geometric Scene Representations.
- Incremental mapping of texture from continuous throughput of sensor data.
- Texture enhancement from multiple images.

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Scene reconstruction
Autonomous driving
Texture mapping

ABSTRACT

Autonomous vehicles have a large number of on-board sensors, not only for providing coverage all around the vehicle, but also to ensure multi-modality in the observation of the scene. Because of this, it is not trivial to come up with a single, unique representation that feeds from the data given by all these sensors. We propose an algorithm which is capable of mapping texture collected from vision based sensors onto a geometric description of the scenario constructed from data provided by 3D sensors. The algorithm uses a constrained Delaunay triangulation to produce a mesh which is updated using a specially devised sequence of operations. These enforce a partial configuration of the mesh that avoids bad quality textures and ensures that there are no gaps in the texture. Results show that this algorithm is capable of producing fine quality textures.

1. Introduction

Autonomous vehicles often have a very large number of sensors mounted on-board. This is due to the need to observe the environment all around the vehicle, but also because vehicles must observe the scene with sensors of different nature. Mainly, sensors are divided into two groups: range sensors and vision based sensors. Sensors of the first group provide 3D measurements of the scene. On the other hand, vision based sensors collect photometric information of the scene. Due to the large number of sensors on-board these vehicles, it is not trivial to combine data from these sensors into a unique representation of the scene. Given that these sensors provide a continuous stream of data over time, and that they are displaced by the movement of the vehicle, then it follows that the representation of the scene must also be dynamic, in the sense that it must evolve to represent novel information collected at later stages of the mission. Note that, given a continuous throughput of images, the most recent image is not necessarily the best image to be used for texture mapping. For example, if the vehicle is moving away from an object, a camera on the rear side of the vehicle will produce images with decreasing quality. Rather, what is required is an algorithm that produces a scene representation at the early stages of a mission (because this might be immediately required for other tasks such as navigation, planning, etc.), but the later on is also capable of evaluating newly acquired images to assess whether or not these images are better than the previously used for mapping the texture. We refer to this as incremental texture mapping.

* Corresponding author at: INESC TEC - INESC Technology and Science, R. Dr. Roberto Frias s/n, 4200-465 Porto, Portugal.

E-mail addresses: m.riem.oliveira@gmail.com (M. Oliveira), vitor@ua.pt (V. Santos), asappa@cvc.uab.es (A.D. Sappa), paulo.dias@ua.pt (P. Dias), amoreira@fe.up.pt (A.P. Moreira).

http://dx.doi.org/10.1016/j.robot.2016.06.009
0921-8890/

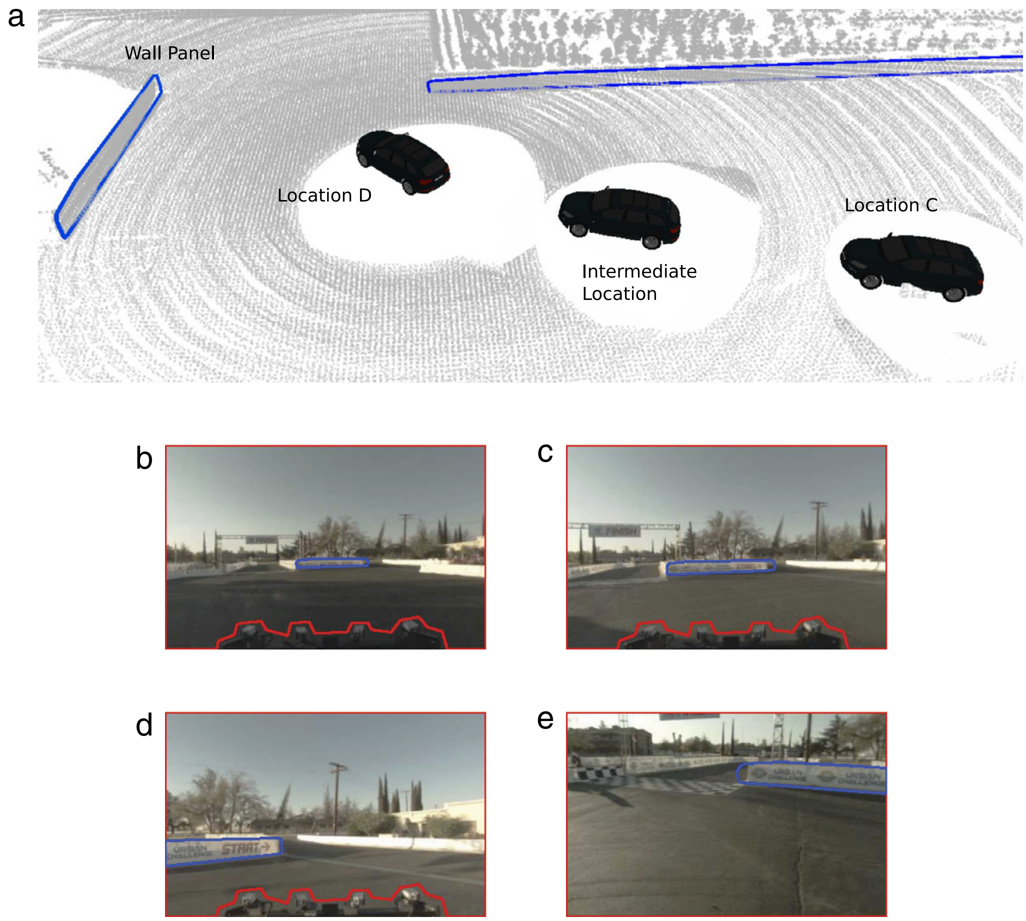

Fig. 1. An example from the MIT data-set: three projections are collected over a period of time and mapped to a wall panel (GPP $k = 4$, in blue): (a) positions of the vehicle at the time each projection is collected; (b) image from front camera, at location C; (c), front camera, intermediate location; (d) front camera, location D; (e) left camera, location D. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In [1], an algorithm for creating and incrementally updating a geometrical representation of the scenario is presented. This work was later extended in [2]. The approach is based on *Geometric Polygonal Primitives* (GPP), and is shown to be capable of providing an accurate geometric description of the scenario. It uses data from range sensors only, and the geometric description changes to accommodate novel sensor data. In this paper we use the results given by the approach described in [2]. This means that we consider that there is, at all times, a geometric description of the scenario which is constantly evolving.

In this paper, we focus on how the vision based sensors can be used to enrich the description of the scenario. In other words, we propose to use the images from the cameras on-board the vehicle to produce texture, which may be added to the 3D description of the environment. Note that, as in the case of the range sensors, the vision based sensors also produce a continuous stream of information which must be integrated in order to create a unique photometric description of the scenario. In this paper, we propose an approach which is capable of incrementally updating texture mapped onto GPPs. The following lines show an example in which the need for incremental texture mapping becomes clear.

For testing and evaluation purposes, we use a data-set from the *Massachusetts Institute of Technology* (MIT) Team, taken from their participation in the DARPA Urban Challenge [3]. A small 40 s sequence was cropped from the MIT data-set. This sequence is referred to as the MIT sequence, and five key locations (A through E where marked in the sequence (see [2] for details). The approach described in [2] produces a description of the geometric structure of the environment observed by the vehicle's sensors. This description is given in the form of *Geometric Polygonal Primitives* (GPP), i.e., a list of polygons. Note that, as pointed out in [2] the geometric description of the scene is dynamic, since it may change whenever novel sensor information is collected.

An example is presented in Fig. 1 where the vehicle travels from location C to location D of the MIT sequence. Images are collected at three locations: location C at mission time t_0, location D at mission time t_2 and an intermediate location between those two at mission time t_1 (Fig. 1(a) shows the vehicle at each location). Consider a camera of index l, that produces an image which may virtually be projected to any GPP (i.e., to one of the polygons that constitute the geometric description of the scene), at any given mission time t. The term projection is defined as an image captured from a camera that can be used to map some texture to one of the polygons contained in the geometric description of the scene, and is denoted as $\mathbf{C}^{\{k,l,t\}}$. The data-set contains five color cameras (see [2,4] for details). Without loss of generality, in this example only images from two cameras are used: front center ($l = 0$) and front left ($l = 3$), and only a single GPP (index $k = 4$) is employed, which corresponds to the wall panel in front of the vehicle (in blue, left side of Fig. 1(a)). Note that, under the constraints defined above, i.e., $k = \{4\}$, $l = \{0, 3\}$ and $t = \{t_0, t_1, t_2\}$, there are a total of six possible projections. However, two of these projections are empty, namely $\mathbf{C}^{\{k=4,l=3,t=t_0\}}$ and $\mathbf{C}^{\{k=4,l=3,t=t_1\}}$. This is because the left camera ($l = 3$) does not see the wall panel ($k = 4$) in the first two locations ($t = t_0$ and $t = t_1$). This can be observed in Fig. 1(a), which shows that the vehicle turns right at location D, and only then the left side camera is pointed in the direction of the wall panel. The images from the remaining four projections are shown