DENSE NON-RIGID VISUAL TRACKING WITH A ROBUST SIMILARITY FUNCTION

Bertrand Delabarre, Eric Marchand

Université de Rennes 1 - IRISA/INRIA Rennes Bretagne Atlantique

ABSTRACT

This paper deals with dense non-rigid visual tracking robust towards global illumination perturbations of the observed scene. The similarity function is based on the sum of conditional variance (SCV). With respect to most approaches that minimize the sum of squared differences, which is poorly robust towards illumination variations in the scene, the choice of SCV as our registration function allows the approach to be naturally robust towards global perturbations. Moreover, a thin-plate spline warping function is considered in order to take into account deformations of the observed template. The proposed approach, after being detailed, is tested in nominal conditions and on scenes where light perturbations occur in order to assess the robustness of the approach.

Index Terms— Visual tracking, non-rigid tracking.

1. INTRODUCTION

Visual tracking is a fundamental step of computer vision. Its field of application is vast and includes for example augmented reality [1], pose estimation [2] or visual servoing [3]. Visual tracking approaches can be divided in several branches. Indeed it is possible to differentiate approaches based on visual features extracted from the images such as keypoints or lines and dense methods also called template-based registration methods relying on a template extracted from a reference image. This paper deals with this latter category. When performing such tracking, the goal is to optimize a registration function representing the difference or similarity between a reference template and the current image. Several works have focused on different registration functions from the most simple, the sum of squared differences (SSD) [4, 5], which compares the luminance of each pixel and is therefore poorly robust to variations of the scene to sophisticated ones such as the mutual information (MI) [6], very robust towards scene perturbations but quite complex to setup. Other functions have also been considered which can be placed in between the two previously mentioned such as the sum of conditional variance [7] or the normalized cross correlation (NCC) [8]. They both are easier to use than the MI and more robust to global illumination variations than the SSD. Those approaches lead to visual tracking algorithms optimizing, for most of them, the parameters of a rigid displacement function (translation, affine motion, homography) in the image frame as in [4, 6, 7] but can also be based on non-rigid displacement functions such as in [9, 10, 11, 12].
The goal of this paper is to introduce a non-rigid visual tracking process robust towards global scene variations such as illumination variations. Our main contribution is to improve the approach described in [7] to take into account non-rigid deformations of the considered template. The use of the SCV allows our algorithm to be naturally robust to global illumination variations which is a frequent problem when performing visual tracking tasks while keeping it simple and fast as the technique is nearly as computationally efficient as the classical SSD-based approach.
The paper is organized as follows. First, the main principles of differential template tracking are recalled and the visual tracking algorithm using the SCV is detailed. Then, the non-rigid warping function used is defined and integrated in the algorithm. Finally, experiments are realized that show how our method is successful in tracking deformable objects in light-varying conditions.

2. DENSE NON-RIGID ROBUST DIFFERENTIAL TEMPLATE TRACKING

Differential template tracking [4] is a class of approaches based on the optimization of an image registration function. They aim at estimating the displacement $\boldsymbol{\mu}$ of a template I^* (that is a set of pixels) in an image sequence. To define the template I^*, the usual method is to extract it from the very first image of the sequence. Then, considering a dissimilarity function f, the problem can be written as:

$$\widehat{\boldsymbol{\mu}} = \arg\min_{\boldsymbol{\mu}} f(I^*, w(I, \boldsymbol{\mu})) \tag{1}$$

where I is the current image of the sequence and $w(I, \boldsymbol{\mu})$ is the template warped thanks to the estimated displacement function (see section 2.4).

2.1. Classical dissimilarity measure: SSD

Most approaches consider the Sum of Squared Differences (SSD) as their registration function. It is the most natural one since it consists in a direct difference between pixel luminosi-

ties in I^* and I. Equation (1) then becomes:

$$\begin{aligned}\widehat{\boldsymbol{\mu}} &= \arg\min_{\boldsymbol{\mu}} SSD(I^*, w(I, \boldsymbol{\mu})) \\ &= \arg\min_{\boldsymbol{\mu}} \sum_{k=0}^{N_{\mathbf{x}}} \left[I^*(\mathbf{x}_k) - I(w(\mathbf{x}_k, \boldsymbol{\mu}))\right]^2 \end{aligned} \quad (2)$$

where $N_{\mathbf{x}}$ is the number of pixels in I^*.

2.2. Considered dissimilarity measure: SCV

Recently, it has been proposed a tracking algorithm based on the sum of conditional variance [7]. The SCV is a template-based dissimilarity function but rather than using the raw template I, as it is the case for the SSD, it is adapted, at each new frame, to match the illumination conditions of the template image I^*, creating an adapted patch $\hat{I}$ thanks to an expectation operator $\mathcal{E}$:

$$\hat{I}(\mathbf{x}) = \mathcal{E}(I^*(\mathbf{x}) \mid I(\mathbf{x})). \quad (3)$$

This operator computes, for each grey level in I, an adapted one which reflects the changes the current template needs to undergo to match the illumination conditions of I^*:

$$\hat{I}(j) = \sum_i i \, \frac{p_{II^*}(i,j)}{p_I(j)} \quad (4)$$

where p_I and p_{II^*} are respectively the probability density function and joint probability density function of I and I^*:

$$\begin{aligned} p_{II^*}(i,j) &= P(I^*(\mathbf{x}) = i, I(\mathbf{x}) = j) \quad (5) \\ &= \frac{1}{N_{\mathbf{x}}} \sum_{k=1}^{N_{\mathbf{x}}} \alpha(I^*(\mathbf{x}_k) - i)\alpha(I(\mathbf{x}_k) - j) \end{aligned}$$

where $\alpha(u) = 1$ if and only if $u = 0$. From this, the probability density function of I is given by:

$$p_I(i) = \sum_j p_{II^*}(i,j). \quad (6)$$

Finally, the difference function is given by:

$$SCV = \sum_{k=1}^{N_{\mathbf{x}}} \left[I^*(\mathbf{x}_k) - \hat{I}(w(\mathbf{x}_k, \boldsymbol{\mu}))\right]^2 \quad (7)$$

which can be expressed within a differential template tracking task:

$$\widehat{\boldsymbol{\mu}} = \arg\min_{\boldsymbol{\mu}} \sum_{k=0}^{N_{\mathbf{x}}} \left[I^*(\mathbf{x}_k) - \hat{I}(w(\mathbf{x}_k, \boldsymbol{\mu}))\right]^2. \quad (8)$$

2.3. Displacement computation

To solve this problem, a classic way to proceed is to adopt an inverse compositional scheme. It consists in searching an increment of displacement $\Delta\boldsymbol{\mu}$ for each new frame and using it to update the global displacement $\boldsymbol{\mu}$:

$$\widehat{\Delta\boldsymbol{\mu}} = \arg\min_{\Delta\boldsymbol{\mu}} \sum_{k=0}^{N_{\mathbf{x}}} \left[I^*(w^{-1}(\mathbf{x}_k, \Delta\boldsymbol{\mu})) - \hat{I}(w(\mathbf{x}_k, \boldsymbol{\mu}))\right]^2 \quad (9)$$

where w^{-1} represents the inverse warping function which is defined for two points $\mathbf{y}$ and $\mathbf{z}$ by:

$$\begin{aligned} \mathbf{y} &= w(\mathbf{z}, \boldsymbol{\mu}) \\ \mathbf{z} &= w^{-1}(\mathbf{y}, \boldsymbol{\mu}) = w(\mathbf{y}, \boldsymbol{\mu}^{-1}). \end{aligned}$$

Given equation (9), a Taylor expansion leads to:

$$SCV(\Delta\boldsymbol{\mu}) \simeq \left[\mathbf{I}^* - w(\hat{\mathbf{I}}, \boldsymbol{\mu}) + \mathbf{J}(\Delta\boldsymbol{\mu})\Delta\boldsymbol{\mu}\right]^2 \quad (10)$$

where $\mathbf{I}^*$ and $\hat{\mathbf{I}}$ are the vectors composed of every points in I^* and $\hat{I}$. $\mathbf{J}(\Delta\boldsymbol{\mu})$ is the Jacobian matrix associated to $SCV(\Delta\boldsymbol{\mu})$, which is defined as:

$$\mathbf{J}(\Delta\boldsymbol{\mu}) = \nabla I^* \frac{\partial w}{\partial \Delta\boldsymbol{\mu}} \quad (11)$$

where ∇I^* is the gradient of I^*. Let us note that the inverse composition scheme allows us to compute $\mathbf{J}(\Delta\boldsymbol{\mu})$ only once since it is constant throughout the tracking phase. From equation (10), a simple optimization step can be iterated where first an increment is computed:

$$\widehat{\Delta\boldsymbol{\mu}} = -\mathbf{J}^+(\Delta\boldsymbol{\mu}) \left[\mathbf{I}^* - w(\hat{\mathbf{I}}, \boldsymbol{\mu})\right] \quad (12)$$

and then the current displacement parameters are updated thanks to:

$$\boldsymbol{\mu} \leftarrow \boldsymbol{\mu} \circ \Delta\boldsymbol{\mu}^{-1} \quad (13)$$

where $\circ$ is the composition operator (which depends on the warp function). Let us also precise that more evolved optimization schemes such as ESM [13] or Levenberg-Marquardt optimizations can also be considered.

2.4. Warping functions

Several warping functions have been considered over the years. From the simplest translation with two parameters [14, 4] to more complex transformations such as affine transformations [5] which adds more freedom of displacement or homographies which traduce the displacement of a plane [15, 14, 4]. The limit of those approaches is that they consider the displacement of a rigid object. If the considered template undergoes deformations, they are not defined to cope with them. To prevent these drawbacks, we propose to use a deformable motion warp. Several warping functions can be considered such as Free Form Deformations [9] or Radial Basis Functions [10, 11]. In this work we propose to use a Thin-plate Spline warp [12].

2.5. Thin-plate Spline warp

Thin-plate spline(TPS) warps belong to the radial basis functions warps. They minimize the bending energy of the surface they parameterize based on a set of control points inducing space coherency constraints. They are based on a thin-plate kernel:

$$\phi_{TPS}(x) = \frac{x^{(4-p)}\ \log(x)}{\alpha} \tag{14}$$

where α and p are parameters that control the freedom given to the deformation. In the remainder of this work, we consider $\alpha = 2$ and $p = 2$, leading to:

$$\phi(x) = \frac{x^2\ \log(x)}{2}. \tag{15}$$

The TPS warp can be seen as an extension of an affine transformation. It is composed of such a warp but adds the possibility of a deformation thanks to the TPS kernel. The considered warping function for our algorithm is then:

$$w(\mathbf{x}, \boldsymbol{\mu}) = \begin{pmatrix} a_0 & a_1 \\ a_3 & a_4 \end{pmatrix} \mathbf{x} + \begin{pmatrix} a_2 \\ a_5 \end{pmatrix} + \sum_{k=1}^{N_p} \begin{pmatrix} w_x^k \\ w_y^k \end{pmatrix} \phi(d(\mathbf{x}, \mathbf{c}_k)). \tag{16}$$

where N_p is the number of considered control points $\mathbf{c}$, w_x^k is the weight of the k^{th} control point along the x axis and $d(\mathbf{x}, \mathbf{y})$ is the euclidean distance between two points $\mathbf{x}$ and $\mathbf{y}$. From equation (16), we can deduce a warp parameter vector of dimension $2N_p + 6$:

$$\boldsymbol{\mu}^\top = (a_0\ a_1\ a_2\ a_3\ a_4\ a_5\ \mathbf{w}_x^\top\ \mathbf{w}_y^\top). \tag{17}$$

Those parameters can be computed thanks to the following system [12]:

$$\begin{pmatrix} \mathbf{K} + \lambda\mathbf{Id} & \mathbf{P} \\ \mathbf{P}^\top & \mathbf{0}_{3\times3} \end{pmatrix} \begin{pmatrix} \boldsymbol{\Omega} \\ \mathbf{A}^\top \end{pmatrix} = \begin{pmatrix} \mathbf{P}' \\ \mathbf{0}_{3\times2} \end{pmatrix}. \tag{18}$$

$\mathbf{A}$ is the matrix composed of the affine transformation parameters:

$$\mathbf{A} = \begin{pmatrix} a_0 & a_1 & a_2 \\ a_3 & a_4 & a_5 \end{pmatrix}. \tag{19}$$

$\boldsymbol{\Omega}$ is the matrix composed of the weights associated to the control points:

$$\boldsymbol{\Omega}^\top = \begin{pmatrix} w_x^0 & ... & w_x^p \\ w_y^0 & ... & w_y^p \end{pmatrix}. \tag{20}$$

$\mathbf{K}$ is given by $\mathbf{K}_{i,j} = \phi(d(\mathbf{c}_i, \mathbf{c}_j))$, $\mathbf{P} = \begin{pmatrix} c_j^x & c_j^y & 1 \end{pmatrix}$ and $\mathbf{P}' = \begin{pmatrix} c'^x_j & c'^y_j & 1 \end{pmatrix}$ where $\mathbf{c}'_k$ is the k^{th} control point once displaced. Let us add that $\lambda\mathbf{Id}$ is a regularization term and that the null matrices are added in order to ensure side conditions. The bigger λ is, the closer the TPS is to an affine transformation. Then, to be able to use this displacement in our algorithm, let us express how the inverse warp parameters are computed. The transfer matrix used in equation (18) is invertible by blocs, leading to:

$$\begin{pmatrix} \boldsymbol{\Omega} \\ \mathbf{A}^\top \end{pmatrix} = \mathbf{E}_\lambda \mathbf{P}' \tag{21}$$

where $\mathbf{E}_\lambda$ is the matrix resulting of the invert of the transfer matrix expressed in equation (18) which is given by:

$$\mathbf{E}_\lambda = \begin{pmatrix} \mathbf{K^{-1}} \left(\mathbf{Id} - \mathbf{P} \left(\mathbf{P}^\top \mathbf{K^{-1}} \mathbf{P} \right)^{-1} \mathbf{P}^\top \mathbf{K^{-1}} \right) \\ \left(\mathbf{P}^\top \mathbf{K^{-1}} \mathbf{P} \right)^{-1} \mathbf{P}^\top \mathbf{K^{-1}} \end{pmatrix}. \tag{22}$$

We can revert the warp from the current control points set $\mathbf{P}^0$ to $\mathbf{P}$ as in [16] giving a new set $\mathbf{P'}$ thanks to the following system:

$$\mathbf{P}' = (\mathbf{O}_p \mathbf{E}_\lambda)^{-1} \mathbf{P}^0. \tag{23}$$

In this case $\mathbf{O}_p$ is a transfer matrix which i^{th} line $\mathbf{O}_{p_i}$ is given by:

$$\mathbf{O}_{p_i} = \left(\phi(d^2(\mathbf{P}_i, \mathbf{P}_0^0)) ... \phi(d^2(\mathbf{P}_i, \mathbf{P}_{N_p-1}^0)) \mathbf{P}_i^\top 1 \right). \tag{24}$$

Finally, let us express the derivative of the warp result with relation to its parameters as needed to compute the Jacobian of our task in equation (11):

$$\frac{\partial w_{TPS}}{\partial \Delta\boldsymbol{\mu}} = \begin{pmatrix} \mathbf{J_A} & \mathbf{J_\Omega} \end{pmatrix} \tag{25}$$

where:

$$\mathbf{J_A} = \begin{pmatrix} x & y & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & y & 1 \end{pmatrix}$$

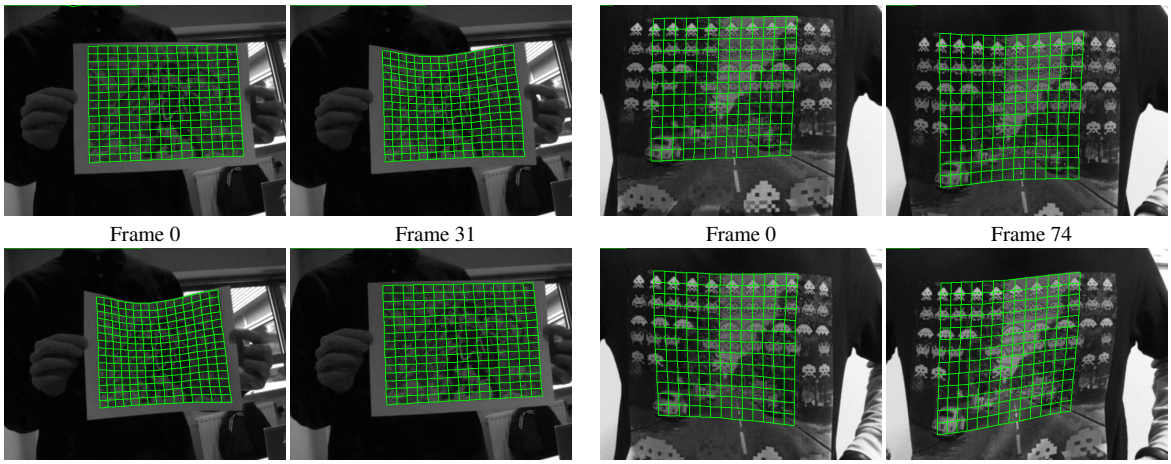
$$\mathbf{J_\Omega} = \begin{pmatrix} \phi(x, c_0) & ... & \phi(x, c_{N_p}) & 0 & ... & 0 \\ 0 & ... & 0 & \phi(x, c_0) & ... & \phi(x, c_{N_p}) \end{pmatrix}$$

3. EXPERIMENTAL RESULTS

To validate our approach, several experiments were realized. For each sequence, a rectangular template is defined on the very first image of the sequence to be registered all along the video sequence. A number of control points is determined and a regular grid matching that number is generated. Assumption is made that all the pixels of the template belong to the same plane, which can be relatively inexact in some cases. The SCV is computed on the current and template images quantified over 64 histogram bins so as to smooth the cost function without losing image information [5, 17].

3.1. Tracking in nominal conditions

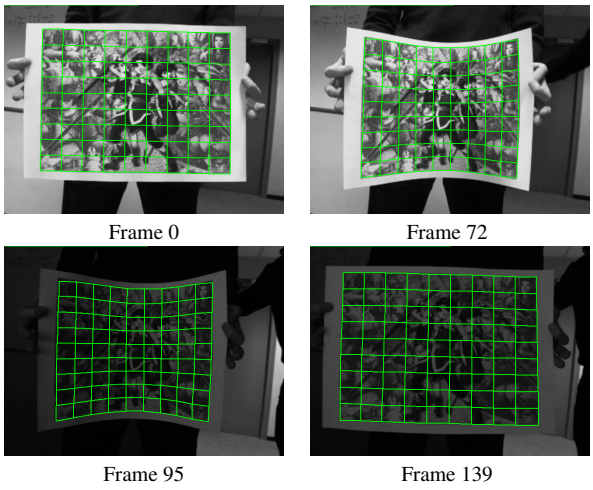
A first experiment was done to validate our approach in nominal conditions. The template is here represented on a piece of paper which is deformed and moved all along the sequence (see figure 1a). We can see that even with a consequent bending of the template the TPS warp matched correctly the deformation and displacement of the paper. Then, the tracking was performed on a sequence where a shirt is deformed. Figure 1b shows that there again, the TPS warp allows our algorithm to cope with that non-rigid displacement.



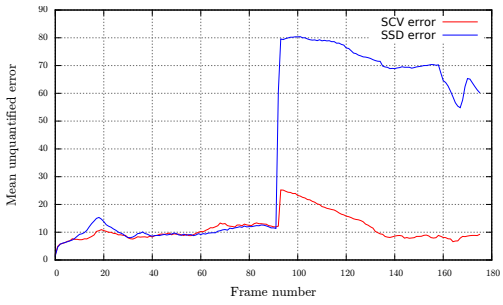
(a) Results of the tracking algorithm in nominal conditions using 256 points. We can see that the shape of the template is correctly matched by the grid of control points.

(b) Results of the tracking algorithm in nominal conditions using 196 points. We can see that the template is correctly deformed when the shirt is moved.

Fig. 1: Results of the proposed algorithm in nominal conditions.



(a) Results of the tracking algorithm in changing conditions using 100 points. We can see that the shape of the template is correctly matched by the grid of control points even when the current illumination is totally different from the reference.



(b) Mean unquantified error by pixel.

Fig. 2: Results of the proposed algorithm in changing conditions.

3.2. Tracking in changing conditions

Our last experiment was realized in order to evaluate our choice of the SCV as a registration function. To do that, we changed the luminosity conditions during the sequence by switching on and off the light of the room. In those conditions, a SSD version of the algorithm was lost as soon as the light changed at iteration 92 (see figure 2b) whereas the SCV completed it without any problems even when brutal changes were applied to the illumination conditions (see figure 2a). We can see on figure 2a the effects of the regularization term on the edges of the template. When the bending is too important, the registration loses in precision along the borders temporarily to keep bending energy low but when the deformation decreases the precision is recovered without any problem. It is also possible to see it on figure 2b where the SCV increases at iteration 95 when bending is too important but gets back to a small value as the bending decreases.

4. CONCLUSION

In this paper we proposed a visual tracking approach that allows to track a deformable template throughout scenes with changing lightning conditions. It consists in an optimization of the sum of conditional variances, which is by definition robust in these conditions. The optimization is done over the parameters of a thin-plate spline warping function to allow non-rigid deformations to happen during the sequence. Future works could consider more robust similarity measures such as for example the mutual information which would allows more robustness towards local variations of the scene.

5. REFERENCES

[1] A.I. Comport, E. Marchand, M. Pressigout, and F. Chaumette, "Real-time markerless tracking for augmented reality: the virtual visual servoing framework," *IEEE Trans. on Visualization and Computer Graphics*, vol. 12, no. 4, pp. 615–628, July 2006.

[2] G. Caron, A. Dame, and E. Marchand, "Direct model-based visual tracking and pose estimation using mutual information," *Image and Vision Computing*, vol. 32, no. 1, pp. 54–63, January 2014.

[3] F. Chaumette and S. Hutchinson, "Visual servo control, Part I: Basic approaches," *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82–90, December 2006.

[4] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *Int. Journal of Computer Vision*, vol. 56, no. 3, pp. 221–255, 2004.

[5] G. Hager and P. Belhumeur, "Efficient region tracking with parametric models of geometry and illumination," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 20, no. 10, pp. 1025–1039, Oct. 1998.

[6] A. Dame and E. Marchand, "Second order optimization of mutual information for real-time image registration," *IEEE Trans. on Image Processing*, vol. 21, no. 9, pp. 4190–4203, September 2012.

[7] R. Richa, R. Sznitman, R. Taylor, and G. Hager, "Visual tracking using the sum of conditional variance," in *IEEE Conference on Intelligent Robots and Systems, IROS'11*, San Francisco, Sept. 2011, pp. 2953–2958.

[8] G. Scandaroli, M. Meilland, and R. Richa, "Improving ncc-based direct visual tracking," in *European conference on Computer Vision*, 2012, ECCV'12, pp. 442–455.

[9] V. Gay-Bellile, A. Bartoli, and P. Sayd, "Direct estimation of nonrigid registrations with image-based self-occlusion reasoning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 87–104, Jan 2010.

[10] N. Arad and D. Reisfeld, "Image warping using few anchor points and radial functions," in *Computer Graphics Forum*. Wiley Online Library, 1995, vol. 14, pp. 35–46.

[11] D. Ruprecht and H. Müller, *Free form deformation with scattered data interpolation methods*, Springer, 1993.

[12] F.L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 6, pp. 567–585, 1989.

[13] S. Benhimane and E. Malis, "Homography-based 2d visual servoing," in *IEEE Int. Conf. on Robotics and Automation, ICRA'06*, Orlando, Fl, May 2006.

[14] B.D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Int. Joint Conf. on Artificial Intelligence, IJCAI'81*, 1981, pp. 674–679.

[15] S. Benhimane and E. Malis, "Integration of Euclidean constraints in template-based visual tracking of piecewise-planar scenes," in *IEEE/RSJ International Conference on Intelligent Robots Systems*, 2006.

[16] V. Gay-Bellile, A. Bartoli, and P. Sayd, "Feature-driven direct non-rigid image registration.," in *BMVC*, 2007, pp. 1–10.

[17] B. Delabarre and E. Marchand, "Visual servoing using the sum of conditional variance," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'12*, Vilamoura, Portugal, October 2012, pp. 1689–1694.