Time to Collision and Collision Risk Estimation from Local Scale and Motion

Shrinivas Pundlik, Eli Peli, and Gang Luo

Schepens Eye Research Institute, Harvard Medical School, Boston, MA
{shrinivas.pundlik,eli.peli,gang.luo}@schepens.harvard.edu

Abstract. Computer-vision based collision risk assessment is important in collision detection and obstacle avoidance tasks. We present an approach to determine both time to collision (TTC) and collision risk for semi-rigid obstacles from videos obtained with an uncalibrated camera. TTC for a body moving relative to the camera can be calculated using the ratio of its image size and its time derivative. In order to compute this ratio, we utilize the local scale change and motion information obtained from detection and tracking of feature points, wherein lies the chief novelty of our approach. Using the same local scale change and motion information, we also propose a measure of collision risk for obstacles moving along different trajectories relative to the camera optical axis. Using videos of pedestrians captured in a controlled experimental setup, in which ground truth can be established, we demonstrate the accuracy of our TTC and collision risk estimation approach for different walking trajectories.

1 Introduction

Time to collision or time to contact (TTC) is a quantity of interest to many fields, ranging from experimental psychology to robotics. Generally, TTC for two bodies in space is the ratio of the distance between them and their relative speed. In the context of video cameras, TTC can be defined as the time required for an object in the real world to reach the camera plane, assuming that the relative speed remains fixed during that time period. While TTC is the ratio of distance and speed, using a pinhole camera model, it becomes equivalent to the computation of the ratio of an object's size on an imaging plane to its time derivative. It has been suggested that analogous processing takes place in the human visual system while performing tasks involving TTC computation, such as avoiding collisions or catching a moving object [1-3]. One of the chief advantages of formulating TTC in terms of object dilation over time is that TTC can thus be obtained entirely from image based data, without having to actually measure physical quantities such as distance and velocity. Consequently, the need for complicated and computationally expensive camera calibration processes, 3D reconstruction of the scene, and camera ego-motion estimation is eliminated. Due to its relative simplicity and computational efficiency, the idea of TTC estimation is ideally suited for real-time systems, where quick decisions have to be made in the face of impending collisions. For this reason, computer-vision based TTC estimation approaches can be useful for obstacle avoidance and collision detection by vehicles or individuals.

G. Bebis et al. (Eds.): ISVC 2011, Part I, LNCS 6938, pp. 732–741, 2011.

The ratio of the object size in the image and its rate of expansion has been previously used for estimation of TTC, for example, computing scale changes over a closed contour using image area moments [4], motion field [5], or affine shape parameters [6]. Accurate initialization is a big challenge in using contours for determining the interest region in the image. This points toward a more general problem of accurately determining object size in the image in order to perform TTC estimation. Image segmentation and object recognition algorithms are complex and thus computationally expensive, and erroneous segmentation can lead to highly inaccurate TTC estimates. To overcome the difficulty of object size determination, TTC estimation could be reformulated in terms of motion field and its derivatives [7, 8], image gradients [9, 10], residual motion from planar parallax [11], scaled depth [12], scale invariant feature matching [13], or solving parametric equations of object motion [14]. A number of previous approaches assume that obstacles are planar rigid bodies in motion relative to the camera along its optical axis. Some approaches, such as [9], are more appropriate when an entire plane moves with respect to the camera and produce inaccurate TTC estimations when a smaller rigid body in front of a static background approaches the camera (a more recent version of this work [10] includes an object segmentation and multi-scale fusion steps which improve TTC estimation results, but still assumes the objects are rigid bodies). Such assumptions fail in situations where semi-rigid obstacles such as pedestrians are involved. Another challenge facing a typical TTC estimation approach is with the case of object motion that is at an angle with the camera axis and not directly towards it. Among the approaches mentioned in this section, very few have dealt with semi-rigid obstacles such as pedestrians, and those that do, show results using only virtual reality scenes [13]. In addition to estimating the TTC accurately for variety of motion trajectories, in applications like collision detection devices, it is also important to determine whether the obstacle moving along a trajectory would indeed collide with the camera platform. This leads to the concept of a collision envelope or a safety zone around the camera, and any object trajectory with a potential to penetrate this zone would then be considered risky.

In this paper, we present an approach for TTC and collision risk estimation in the case of semi-rigidly moving obstacles using feature points. The novelty of our approach is that the computation of TTC is based on aggregation of local scale change and motion information to obtain a global value for an object. In addition to TTC, the approach can also predict the collision risk for a given object trajectory relative to the camera. This collision risk is the probability of collision, and can be tailored to different collision warning scenarios by setting an acceptable threshold for different applications. We demonstrate the effectiveness of our approach by estimating TTC and collision risk using videos of pedestrians walking along different trajectories captured from an uncalibrated camera.

Processing in our approach proceeds in the following manner. Detection and tracking of feature points is performed on the input image sequence. This provides us with the sparse motion information present in the scene. Scale change computation is performed in the neighborhood of each point feature, and a set of feature points where there is an increase in the local scale between two frames of a sequence is obtained. For this set of feature points, an affine motion model is fitted, leading to a group of features associated with a potential obstacle. The use of feature points and affine motion model provide flexibility to represent a semi-rigidly moving obstacle. From the

features associated with the obstacle, TTC and collision risk is estimated. The following sections describe the details of our approach and the experimental results.

2 TTC Estimation Using Feature Points

Feature point tracking forms the basis of our approach [15]. Feature points are effective in quantifying the motion in a local image patch. Unlike dense motion field computation, or Scale Invariant Features (SIFT) [16], feature points can be detected and tracked in an efficient manner. Another advantage of using feature points over a dense motion field is that textureless regions, where motion estimates tend to be erroneous, can be avoided. In real-world situations, there is usually enough texture to obtain a sufficient number of reliable feature point trajectories, thus providing some tolerance from the loss of some feature points due to scene changes over time.

For a given image sequence (320x240 pixels), we detect and track features over a block of b frames. The detected features are based on the algorithm described in [15] and are known to be more suitable for tracking over an image sequence as compared to Harris corners. We use the pyramidal implementation of the Lucas-Kanade algorithm for feature tracking [17]. The pyramid levels are set at 3. Only those feature points with quality above a threshold value are selected for tracking. The threshold used in this work is set as 5% of the highest quality feature point selected in a given image. The size of the feature window is set at 5x5. Let $p_i^{(j)}$ be the i^{th} feature point in the j^{th} frame and n be the total number of feature points tracked through b frames. Once the features are detected and tracked, we compute feature point neighborhood using Delaunay triangulation. For every feature point, we now obtain a set of immediate spatial neighbors in the image that share an edge of the triangulation with that feature. Let this set of feature points in the local Delaunay neighborhood for $p_i^{(j)}$ be denoted by $D(p_i^{(j)})$.

Once the feature points are tracked and a local neighborhood is established, the next step is to compute local scale change for each feature point. As the object approaches the camera, it gradually increases in size. Consider a rigid body moving along the camera's optical axis. As it approaches the camera, the distance between neighboring feature points increases. This is the low-level manifestation of change in scale. Even if the motion does not take place strictly along the optical axis, the amount by which the feature points dilate could be significant (though not quite as much as in the previous case). We use this observation to compute the local scale change between two frames of a sequence, and it is given by

$$s_i^{(j)} = \frac{\sum_{p_k^{(j)} \in D\left(p_i^{(j)}\right)} \left(\left\| p_i^{(j)} - p_k^{(j)} \right\| - \left\| p_i^{(j-b)} - p_k^{(j-b)} \right\| \right)}{\sum_{p_k^{(j)} \in D\left(p_i^{(j)}\right)} \left\| p_i^{(j-b)} - p_k^{(j-b)} \right\|}, \tag{1}$$

where $s_i^{(j)}$ is the normalized local scale change value for the i^{th} feature point in the j^{th} frame. Hence, for each point we obtain a positive scale change value when the neighborhood of the point feature expands and a negative value when it shrinks. We threshold

the local scale values to obtain a set of feature points that show a significant degree of increase in scale. Let this threshold be denoted by λ_s, such that we are interested in feature points for which $s_i^{(j)} > \lambda_s$ ($\lambda s = 0.1 max(s_i^{(j)})$ was used for this work).

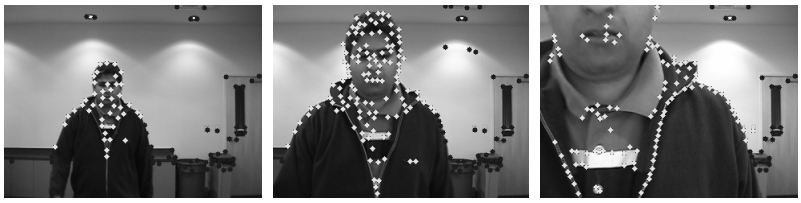

Fig. 1. Frames 50, 95, and 120 of a sequence in which a person walks approximately along the optical axis. Based on affine motion and local scale change, the point features are grouped as those belonging to the moving person (white diamonds) and the background (black asterisks).

Once we obtain a set of feature points undergoing local scale increase, we process the motion information associated with these feature points. First, a Random Sample Consensus (RANSAC) algorithm is applied to this set of feature points to perform grouping based on affine motion. For a feature group, this also serves as an outlier rejection step. The next step is to include ungrouped feature points (excluded in the previous scale based thresholding step) into the newly-formed feature groups based on their motion compatibility. This step results in a stable set of feature points, denoted by $F^{(j)}$, that are associated with a moving body. Fig. 1 shows three frames of our test sequence in which a pedestrian approaches the camera. The feature points $F^{(j)}$, associated with the walking person and the background are overlaid. Although the above approach is designed to handle multiple groups of feature points, in this paper we focus on a single dominant group that belongs to the pedestrian's body for TTC computation. The TTC value for each point feature neighborhood associated with the moving body is now given by $t_i^{(j)} = 1/s_i^{(j)}$. For a feature group, the TTC value $t^{(j)}$ (for the j^{th} frame), is the median of the TTC values obtained from all the feature points belonging to the group. The TTC value obtained here is in terms of frames remaining before collision. It could be converted to seconds using the video frame rate. For this work, we capture test videos at 30 frames per second.

One way of evaluating the collision risk for an obstacle would be to assign a threshold value for TTC such that estimates below this value represent a high collision risk. But this does not take into consideration the point of intersection of the obstacle trajectory with the camera plane, and it could be far from the camera center to be considered a collision risk. In order to evaluate the risk posed by collisions from different obstacle trajectories, we propose an additional measure, which we call the collision risk factor. This factor is based on the ratio of the local scale change and local motion between two frames. For an obstacle moving along the camera optical axis this ratio would be high, while for motion along trajectories that are at increasing angles with the camera optical axis this ratio would be smaller. For the i^{th} point feature, the ratio of local scale change and motion is given by $d_i^{(j)} = s_i^{(j)} / v_{i,}$ where

$v_i = \left\| p_i^{(j)} - p_i^{(j-b)} \right\|$ is the magnitude of the motion vector of the i^{th} feature point between the j^{th} and $(j\text{-}b)^{\text{th}}$ frame. The collision risk associated with an object is the local scale change to motion ratio for the corresponding feature group. For an obstacle in the j^{th} frame of an image sequence, the collision risk factor is given by

$$c^{(j)} = \frac{1}{m} \sum_{\forall p_i^{(j)} \in F^{(j)}} d_i^{(j)}, \tag{2}$$

where m is the number of features belonging to the moving object. The collision risk factor in Eq. (2) represents the idea that as an object approaches the camera along (or close to) the optical axis, large local scale change values lead to larger values of the ratio d as compared to other obstacle trajectories (where the magnitude of lateral motion is typically larger than the local scale change). The value of $c^{(j)}$ in such situations increases while the TTC value decreases. Hence, both collision risk factor and TTC when combined present a robust measure of collision risk. It should be noted that collision risk is a purely image-based quantity (no physical units), and along with TTC it can be used for issuing collision warnings by setting an appropriate threshold.

3 Experimental Results

We present experimental results of testing our approach using videos of 2 pedestrians walking along different predefined trajectories, acting as potential obstacles for which TTC and collision risk are estimated. The goal of such an experimental setup is to simulate real world conditions as closely as possible without resorting to the use of synthetic sequences, while obtaining the ground truth for quantitative comparison.

3.1 Experimental Setup

Fig. 2-(a) shows the detailed schematic of the experimental setup. It consists of two cameras capturing videos in a large room, approximately 20x80 feet. Camera 1 is set up at location C1 along baseline1 to capture the videos to be processed by our TTC estimation algorithm. Another baseline (baseline2) is established 204 inches (17 feet) away from baseline1. A person walks along the 11 trajectories defined by lines R5-L5 to L5-R5, passing through a center point C, which is about 8.5 feet away from Camera 1. On each side of the optical axis of Camera 1, the five trajectories make increasing angles of 10, 20, 30, 37.5, and 45 degrees with the center line C-C1 (see Fig. 2-(a)). While capturing the videos, the trajectory lines were not explicitly drawn on the floor. Only the points marked on the two baselines and the center point C were placed on the ground and these markers were used for guidance by the pedestrians. In order to obtain the ground truth world positions of the pedestrians with respect to Camera 1, we captured profile views simultaneously from Camera 2 (both the cameras are synchronized). The perpendicular distance between the line C-C1 and Camera 2 was about 58 feet. A larger distance minimizes the effect of depth for different trajectories and ensures a sufficiently large camera FOV to cover the entire sequence of walks. All the physical distances in this setup were obtained from a standard measuring tape.

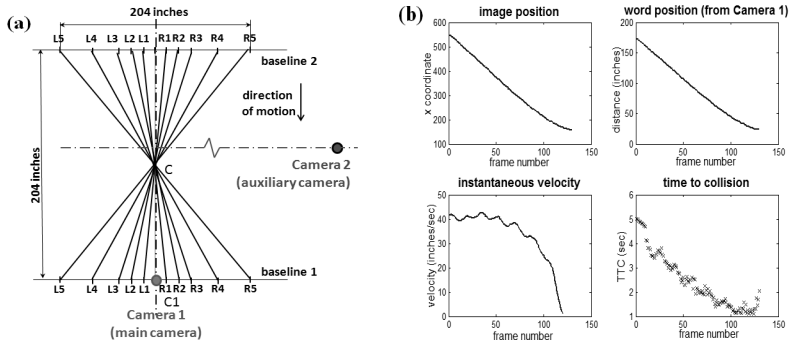

Fig. 2. (a): A schematic (top view) of the experimental setup. (b): For a pedestrian on the C-C1 trajectory, plots of pedestrian position as seen from Camera 2, world distance from Camera 1, instantaneous velocity (computed over 10 frames), and the derived ground truth TTC values.

In order to obtain a value of image distance for the corresponding world distance, we measured known distances on the centerline C-C1. We also determined the FOV and the angle per pixel for Camera 2. Based on the known and computed quantities from the experimental setup, and the a priori knowledge of the trajectory being currently traced, we derived a model to estimate the position of the pedestrian in the real world with respect to Camera 1 using Camera 2 image positions. The mathematical details of this procedure are left out of this paper due to space constraints. We tested our ground truth measurement procedure by estimating the world locations (with respect to Camera 1) of 10 known world points in the experimental setup. We obtained a maximum error of about 5 inches and an average error of about 2.6 inches. The estimation error progressively increased for points with increasing distance from the centerline and baseline1. For the locations where ground truth measurement would significantly affect the accuracy of the validation process (such as points near baseline1), the mean error in ground truth measurement was 1.5 inches.

We manually labeled the location of the pedestrian in images obtained from Camera 2 by marking 6 points along the upper body profile and computing the mean x coordinate value. The ground truth computation procedure outputs the corresponding world location and instantaneous velocity (obtained by differentiating the distance over 10 frames or 1/3 of a second). Once the distance from baseline1 and the instantaneous velocity are known, ground truth TTC values for each frame of the sequence can be computed. Fig. 2-(b) shows the plots of the intermediate quantities involved in computing the ground truth TTC values. The figure shows that the image and world positions for C-C1 trajectory are linear for the most part and the velocity recorded is approximately constant (40 inches/sec.). The small variations in the velocity are due to variable walking speed, error in labeling procedure, and due to error in the ground truth computation procedure. The velocity (along the walking direction) decreases sharply in the later part of the trajectory because the pedestrian slows down as he approaches the camera, while the TTC values (time to baseline 1) increase for this phase of the walk.

3.2 TTC Estimation Results

TTC estimation results of our algorithm along with the corresponding ground truth values for three trajectories (out of possible 22 for both the pedestrians) are shown in Fig. 3-(a),(b), and (c). Each plot is also superimposed with some frames of the corresponding sequence to show the change in the appearance of the pedestrian over time. The plot in Fig. 3-(a) shows the case where the person approaches the camera head-on along the C-C1 trajectory. Fig. 3-(b) and (c) show the results when the pedestrian walks with an angle of approximately 10 and 30 degrees with the optical axis, respectively. The estimated TTC values follow the same trend as the ground truth values. More importantly, at lower TTC ranges (as the relative distance between the pedestrian and the camera decreases), the estimates follow the ground truth more closely. This is a desired property because the estimates need to be more accurate when the obstacle is perceived to be close to the camera. Our algorithm can also handle variable relative velocity. At the very end of the C-C1 trajectory, the person slows down before coming to a halt. The estimated TTC values start increasing corresponding to this change. The plot in Fig. 3-(d) shows the mean collision risk per feature point (belonging to the pedestrian) measured over the last 1 second of the run (i.e., 1 sec before the pedestrian stops just a few inches away from the camera) for all the trajectories for both pedestrians. The curves show increased risk for the C-C1 trajectory, as desired.

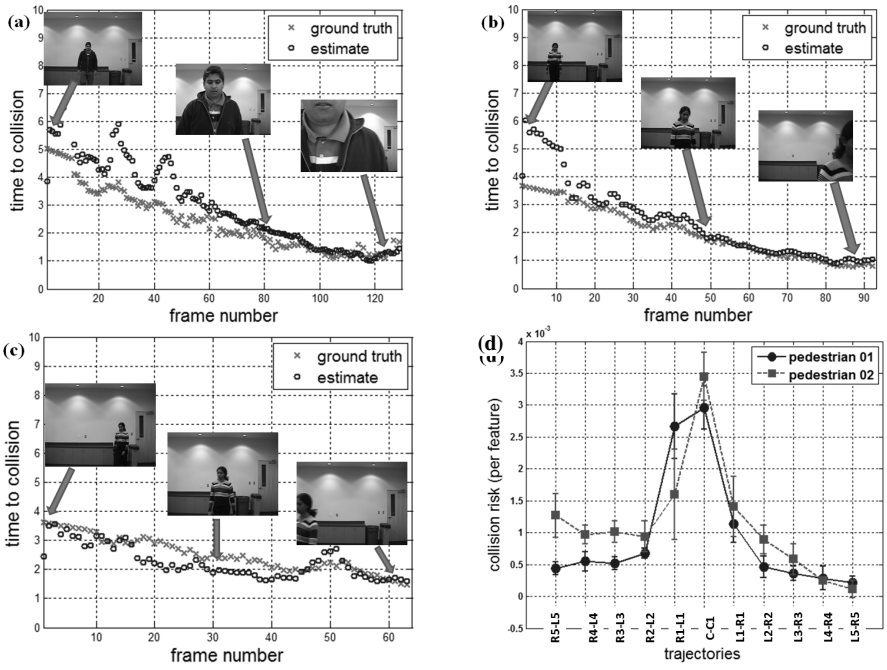

Fig. 3. Plots of TTC estimates (smoothed with a temporal window of 3 frames), the ground truth values and three representative frames demonstrating pedestrian position for trajectories C-C1 for pedestrian 1 **(a),** L1-R1 **(b)** and R3-L3 **(c)** for pedestrian 2. A plot of collision risk associated with the obstacle for the last 1 second for each trajectory is shown in **(d)**.

Also, the values decrease progressively on either side of the central line as the trajectory angles increase, indicating lower risk of collision when an obstacle moves along these trajectories. Since the collision risk measured in Eq. (2) is normalized by the number of feature points associated with the object, the plot also shows that the collision risk is not directly dependent on the size of the obstacle in the image (area covered in the image as the pedestrian come close to the camera). Also, Fig. 3 shows that even though the TTC for different trajectories converge at relatively close values at the end of the run, the corresponding collision risk values are significantly different.

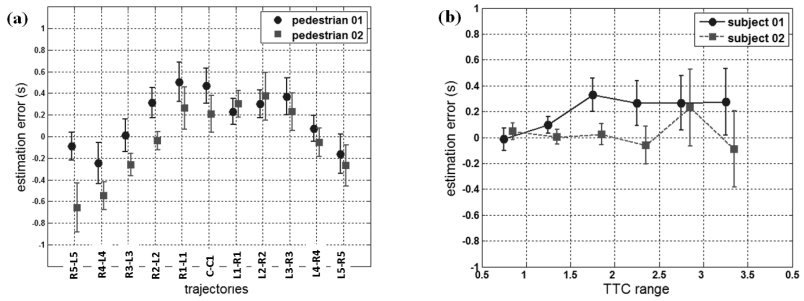

Fig. 4. (a): Plot of TTC estimation error (mean and std. error) per trajectory. **(b)**: Plot of TTC estimation error (mean and std. error) for all trajectories for different TTC ranges.

Fig. 4 shows the performance of the proposed approach by providing some quantitative analysis regarding the error in TTC estimation. Fig. 4-(a) shows the mean estimation error and the standard error for each trajectory. It can be seen that for the trajectories closer to the optical axis of Camera 1, the TTC is overestimated (mean error is about 0.5 seconds). It should be noted that we left out TTC values that were more than 3.5s while generating Fig. 4, as these translate to a distance between the camera and the pedestrian larger than about 13 feet. The estimates at locations farther away from the camera tend to be less reliable since the apparent scale change is smaller compared to those obtained at locations closer to the camera. To get a better idea about the performance of our algorithm, Fig. 4-(b) shows the mean estimation error and standard error for all the trajectories for different TTC ranges. It can be seen that the mean and standard error for the estimations are lower for lower TTC ranges for both the pedestrians and they become progressively larger for higher TTC ranges (i.e., larger distance from the camera). This plot quantifies our earlier claim that estimates of our approach are more accurate as the obstacle nears camera. For comparison, we also implemented the direct method presented in [9] for the case of motion along the optical axis of a plane perpendicular to the optical axis (with 8x8 block averaging). For our experimental setup, only the C-C1 trajectory loosely fits both the criteria. Combining the error in the TTC estimates from both pedestrians, the mean and standard error of 4.9s and 1.3s was obtained, as compared with our values of 0.3s and 0.4s respectively (seen in Fig. 4-a).

4 Conclusions

We have presented an approach for TTC and collision risk estimation from local scale change and motion information in the scene using feature tracking. The proposed approach can accurately estimate TTC and collision risk for semi-rigid obstacles moving along different trajectories relative to the camera's optical axis with varying speeds, especially when they are close to the camera. The collision risk factor, which is insensitive to the obstacle's size on the image, is highest for the walking trajectory along the camera optical axis and it progressively reduces for the others. Even though the use of feature points affords us a lot of flexibility in terms of handling semi-rigid motion along arbitrary trajectories, there are certain limitations of the current approach. Brightness constancy assumption is implicit in feature tracking. Also, the approach is not robust in tackling scenarios where the pedestrians suddenly appear in the scene very close to the camera. Current experimental setup did not allow us to test a variety of different potential obstacles or moving camera scenarios (along with in-plane and out-of-plane rotations) since our goal was to thoroughly evaluate the TTC and collision risk estimation using controlled experimentation. In spite of this, the results presented here show that our approach has the potential to be effective in complex real-world scenarios. Future work includes extension of the current algorithm to handle more complex real world scenarios involving multiple obstacles, moving camera, and variable lighting conditions.

Acknowledgements. This work was supported in part by DoD grant W81XWH-10-1-0980, DM090201 and by NIH grant R01 EY12890.

References

1. Lee, D.N.: A theory of the visual control of braking based on information about time-to-collision Perception 5, 437–459 (1976)
2. Tresilian, J.R.: Visually timed action: time-out for 'tau'? Trends in Cognitive Sciences 3, 301–310 (1999)
3. Luo, G., Woods, R., Peli, E.: Collision judgment when using an augmented vision head mounted display device. Investigative Ophthalmology and Visual Science 50, 4509–4515 (2009)
4. Cipolla, R., Blake, A.: Surface orientation and time to contact from divergence and deformation. In: Sandini, G. (ed.) ECCV 1992. LNCS, vol. 588, pp. 187–202. Springer, Heidelberg (1992)
5. Ancona, N., Poggio, T.: Optical flow from 1d correlation: Application to a simple time to crash detector. International Journal of Computer Vision 14, 131–146 (1995)
6. Alenya, G., Negre, A., Crowley, J.L.: A Comparison of Three Methods for Measure of Time to Contact. In: IEEE/RSJ Conference on Intelligent Robots and Systems, pp. 1–6 (2009)
7. Meyer, F.G.: Time-to-collision from first order models of the motion field. IEEE Transactions on Robotics and Automation 10, 792–798 (1994)
8. Camus, T.A.: Calculating time-to-contact using real time quantized optical flow. Max-Planck-Institut fur Biologische Kybernetik Technical Report (1995)

9. Horn, B.K.P., Fang, Y., Masaki, I.: Time to Contact Relative to a Planar Surface. In: IEEE Intelligent Vehicle Symposium, pp. 68–74 (2007)
10. Horn, B.K.P., Fang, Y., Masaki, I.: Hierarchical framework for direct gradient-based time-to-contact estimation. In: IEEE Intelligent Vehicle Symposium, pp. 1394–1400 (2009)
11. Lourakis, M., Orphanoudakis, S.: Using planar parallax to estimate the time-to-contact. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 640–645 (1999)
12. Colombo, C., DelBimbo, A.: Generalized bounds for time to collision from first order image motion. In: IEEE International Conference on Computer Vision, pp. 220–226 (1999)
13. Negre, A., Braillon, C., Crowley, J.L., Laugier, C.: Real time time to collision from variation of intrinsic scale. In: Proceedings of the International Symposium on Experimental Robotics, pp. 75–84 (2006)
14. Muller, D., Pauli, J., Nunn, C., Gormer, S., Muller-Schneiders, S.: Time to Contact Estimation Using Interest Points. In: IEEE Conference on Intelligent Transportation Systems, pp. 1–6 (2009)
15. Shi, J., Tomasi, C.: Good Features to Track. In: IEEE Conference On Computer Vision And Pattern Recognition, pp. 593–600 (1994)
16. Lowe, D.: Distinctive image features from scale invariant keypoints. International Journal of Computer Vision 60, 75–84 (2004)
17. Bouguet, J.Y.: Pyramidal implementation of the lucas-kanade feature tracker (2000)