Intuitive Visualization of Vehicle Distance, Velocity and Risk Potential in Rear-View Camera Applications

Christoph Roessing[1], Axel Reker[1], Michael Gabb[2], Klaus Dietmayer[2] and Hendrik P. A. Lensch[3]

***Abstract*— Many serious collisions on highways happen while changing lanes. One of the main causes for these accidents is the driver's incorrect assessment of the current rear traffic situation. To support the driver, we propose a framework to intuitively visualize distance, speed and risk potential of approaching vehicles in a rear-view camera application. The proposed visualization techniques are based on color coding, artificial motion blur and depth-of-field rendering, which are motivated by sensory effects of the human eye and interpreted intuitively by the human visual system. The impact on the human assessment of the moving speed of an object rendered with artificial motion enhancement is evaluated in a user study. The required distance and motion estimation of the vehicles are extracted out of monocular video images, by combining lane recognition, vehicle detection and segmentation machine vision algorithms.**

I. INTRODUCTION

Every year many people die in traffic accidents, caused by drivers wrong judgment of the rear traffic situation. Modern vehicles are equipped with forward-looking cameras and radar sensors to support the driver in various driving situations (night vision [1], lane detection [2], etc.). To support the driver in his assessment of the rear traffic, vehicles could be equipped with backward-looking cameras, as well. By incorporating machine vision algorithms, the position and velocity of following vehicles and the risk potential of a collision can be evaluated and conveyed to the driver.

However, overlaying all available information on distance, speed and danger potential of all following vehicles in a rear-view camera image could overextend the driver. Therefore we propose several visualization techniques, which convey the critical information in an undisturbing, yet distinct manner (see Figure 1).

II. RELATED WORK

Human beings perceive their environment mainly with their eyes, by means of a visual system, with a high spacial resolution ($0,6'$ arc) and stereo capabilities for perceiving depth [3]. However, the mental representation of the environment would not be as precise without the pre-knowledge and interpretation of the visual cortex. The human visual system is able to obtain additional information about motion and distances by subconsciously analyzing the 2D images on the retina. This may even trigger a sensation of motion and depth while looking at 2D still images.

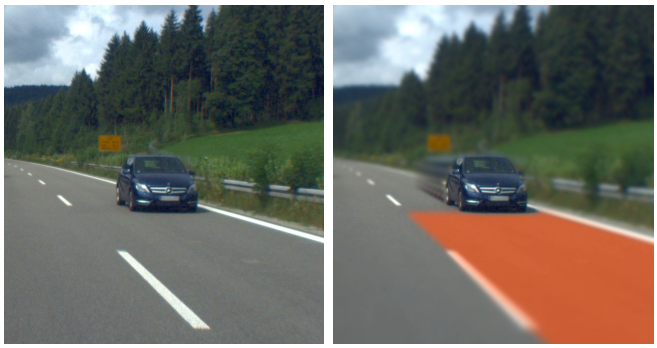

Fig. 1. Original image (left) vs. artificial motion blur, depth-of-field and risk potential visualization to support the drivers assessment of the traffic situation behind (right)

A. Monocular Depth Perception

Held *et al.* [4] studied the impact of artificial depth-of-field renderings in 2D images on the human depth perception. They discovered, that a strong sensation of depth can be induced in the human perception by applying an artificial depth-of-field on sharp input images. By exaggerating this monocular depth cue, they were even able to induce a sensation of close distances on aerial perspective Pictures and thereby a miniaturization effect in the human perception. These findings indicate, that an artificial depth-of-field is able to convey distance information in 2D still images.

Roessing *et al.* [5] proposed a realtime framework to enhance several monocular depth cues in sharp input images based on a supplemental depth map. They applied depth-of-field, in-focus local contrast enhancement and added shadows at depth discontinuities. These image manipulations enhance the perception of depth in 2D images and draw the viewers attention towards selected image regions and objects.

B. Motion Blur

Motion blur is usually known as side effect of capturing images of a moving object with a long exposure time. A similar sensory effect occurs in the human retina: the impact of photons activate a chemical process which triggers a stimulus in the optic nerve. The stimulus remains until the chemical process is reverted. Dependent on the brightness of the stimuli, these afterimages last about 0.04sec [6]. In case of a moving object, multiple afterimages are superimposed and result in a perceivable motion blur.

The visual cortex processes this motion blur subconsciously and induces the perception of a moving object in the mental representation of the scenery. Accordingly, a perception of

[1]C. Roessing and A. Reker are with
Daimler AG Research and Development, Ulm, Germany
firstname.lastname at daimler.com

[2]M. Gabb and K. Dietmayer are with the Department of Measurement, Control and Microtechnology, driveU / University of Ulm, Ulm, Germany
firstname.lastname at uni-ulm.de

[3]H. Lensch heads the Computer Graphics Group,
Eberhard Karls Universitaet Tuebingen, Tuebingen, Germany
hendrik.lensch at uni-tuebingen.de

motion can even be induced by looking at a still image which shows a motion-blurred object [7].

To render a plausible motion blur, the 3D structure and the motion vectors have to be known. Hence, the artificial rendering of motion blur was first introduced in 3D rendering, wherein the 3D environment and motion vectors are fully known. Korein and Badler [8] proposed to superimpose multiple renderings of a moving object with a fading intensity gradient to generate a discrete motion blur (stroboscope). To render a photorealistic motion blur, Potmesil and Charkravarty [9] extended this technique by accumulating the object movement in the image plane to a path and performed a convolution of every sample point along this path.

In real video images, the motion vector of the visible objects in consecutive frames is usually not known. Therefore Brostow and Essa [10] incorporated machine vision algorithms to calculate a dense optical flow between consecutive frames and rendered photorealistic motion blur in stop motion animations and real video sequences. Kim and Essa [11] adapted the concept of non-photorealistic motion blur (mostly known from speedlines in comics) for real video images. Even such an over-exaggerated and unnatural display of motion blur is subconsciously processed and interpreted by the visual system, thus inducing the perception of fast movement.

C. Machine vision and segmentation

To render artificial depth-of-field or motion blur, the spacial arrangement and motion vectors within the video image have to be known. Usually, machine vision algorithms, like stereo vision [12] or dense optical flow are used to generate the input for the rendering pipeline [5] [11]. Instead of a full 3D reconstruction, which would cause a large computational cost, we exploit that the scenery is predefined to highway scenarios and the occurring moving objects are confined to vehicles. Thus monocular machine vision algorithms are sufficient to gather the required supplemental data (see Section III).

LANE RECOGNITION The lane recognition reconstructs the roadway course by detecting the lane markings in a 2D video image. In this framework the algorithm described by Kaeppler *et al.* [2] provides the road course estimation in form of a clothoid $y_{lane}(x)$. The additional information to calculate the clothoid (lane width w, lateral shift y_{shift} and current orientation ψ_{lane} of the vehicle) are provided, as well.

$$y_{lane}(x) = \pm 0.5w - y_{shift} - \psi_{lane}x + \frac{1}{2}y_0x^2 + \frac{1}{6}y_1x^3 \quad (1)$$

ODOMETRY To estimate the ego-motion of the vehicle, the odometry data measured by the wheel angle sensors and the yaw rate sensors are fused by a Kalman filter [13] to estimate the ego-motion within the last step t in x direction $\Delta x(t)$ and turn rate $\psi_v(t)$.

VEHICLE DETECTION A framework based on the Boosted Cascade [14],[15],[16] is used to obtain the 2D position and size of approaching vehicles in input images. It follows an approach called "Detection by Classification" whereby a trained classifier is used to separate rectangular image subwindows (ROIs) that show vehicles from those ROIs that show background. To detect vehicles, input images are densely scanned with ROIs in all possible locations and sizes. Only ROIs reaching a classification score above a chosen threshold are marked as vehicles. As a result, multiple ROIs with nearly the same size and location result in a positive detection at a single vehicle location. A final non-maximum suppression step based on the Mean Shift approach [17] is used to combine these multiple ROIs into a single ROI per vehicle. Due to the ROI scan quantization, even combined and averaged ROIs never match the real vehicle position and size perfectly [18]. Additionally, the average detection rate is only about 95%. Hence, in subsequent frames the classifier is unstable with respect to re-detection, size and position of the vehicle.

GRAB CUT SEGMENTATION Graph-based segmentation techniques have shown to be superior compared to other approaches [19]. While the original Graph Cut algorithm [20] requires an initial pixel labeling provided by the user, Rother and Kolmogorov [21] proposed an enhanced version that greatly reduces the user input for initial scene labeling. Their key idea is an iterative Graph Cut implementation in conjunction with a stepwise label refinement. This iterative technique requires only a bounding box of the foreground object as initial labeling for the segmentation. Thus it can be used without user interaction to segment vehicles out of the image, by generating the initial bounding box from the vehicle detection results.

Based on these insights we propose a driver assistance system that incorporates the sensory input from monocular camera video streams which are analyzed to reconstruct the captured spacial arrangement. Based on this reconstruction, we demonstrate how various selective visualization techniques applied to the rear-view camera stream can alter the perceived relative speed of trailing vehicles.

III. RECONSTRUCT THE 3D SURROUNDING

To generate plausible and natural visualizations, the 3D structure and motion of the visible objects in the rear-view video stream have to be reconstructed. For this purpose the gathered sensor data is fused into one common 3D coordinate frame as described in (ISO 8855). All objects that have been transformed in this coordinate frame form the vehicle's environment model $M = [p_1, p_2, ...]$. Within this paper, it is assumed that the vehicle is moving on a flat plane, which might cause errors in the 3D reconstruction and therefore in the visualization, as well. Nonetheless, our results show that the induced errors in the rendered output is negligible for most scenery (see Section IV).

A. Lane detection

To reconstruct the road course, an extrinsically and intrinsically calibrated forward-looking camera captures the scenery ahead and detects the roadway markings (see Section II-C). The lane markings could be detected with the rear-view camera as well, but depending on the installation location of the camera the markings might be occluded by the vehicle itself. The environment model M is built by adding the current coordinates $y_{lane}(0)$ of the detected lane markings next to the rear axes $(x = 0)$ (see equation (2)).
At the following time step $(t+1)$, one has to assume, that the vehicle has moved in x direction (Δx) and rotated around the z axis (ψ_v), which is estimated by the odometry filter (see Section II-C). To keep the environment model consistent with the visible rear-view camera image, the coordinates in the environment model for the new time step $M(t+1)$ have to be translated in x (T_x) and rotated in z (R_z), inverse to the ego-motion of the vehicle (see equation (3)). To visualize the results of the lane recognition, the 3D lane markings are re-projected into the rear-view camera image. The projection matrix P is built out of the intrinsic $\big($focal length f_0 and center of distortion $u_0, v_0\big)$ and extrinsic $\big($yaw (ψ_c), pitch (θ_c) and roll (ϕ_c) angles and 3D translation $\vec{(}v)\big)$ camera calibration (see equation (4)).

$$M(t) = \begin{pmatrix} x_0 & x_1 & x_2 & \dots & 0 \\ y_0 & y_1 & y_2 & \dots & y_{lane}(0) \\ z_0 & z_1 & z_2 & \dots & 0 \\ 1 & 1 & 1 & \dots & 1 \end{pmatrix} \tag{2}$$

$$M(t+1) = M(t) \cdot T_x(-\Delta x(t+1)) \cdot R_z(-\psi_v(t+1)) \tag{3}$$

To visualize the detected lanes, the re-projected roadway markings are connected to a closed surface (see Figure 2).

$$P = \underbrace{\begin{bmatrix} f_0 & 0 & u_0 & 0 \\ 0 & f_0 & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{intrinsic} \underbrace{\cdot \underbrace{R_z(-\psi_c)}_{yaw} \cdot \underbrace{R_y(-\theta_c)}_{pitch} \cdot \underbrace{R_x(-\phi_c)}_{roll} \cdot T(-\vec{v})}_{extrinsic} \tag{4}$$

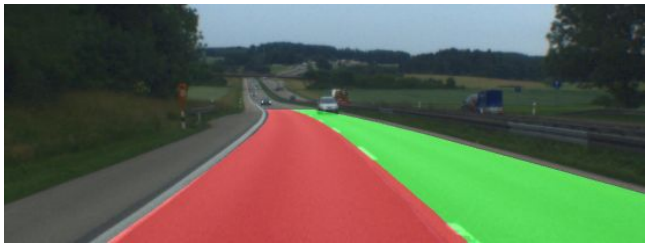

Fig. 2. Re-projected overlay of the detected and propagated lanes in the rear-view camera image

B. Vehicle detection and tracking

The colorization of the lanes and the road course may support the driver in their assessment of the rear traffic. However, the risk potential arises from the following vehicles and the driver's wrong estimation of their relative speed and distance. Our framework extracts this information from the rear-view camera image: A robust classifier based vehicle detection algorithm for single images was presented in Section II-C. We extend this detection method with the Grab Cut segmentation algorithm to track the vehicles and to obtain a coarse, but robust distance and velocity estimation.

VEHICLE DETECTION To speed up the classifier and to suppress faulty detections in vehicle-resembling structures, the algorithm is restricted to the re-projected road course. The classifier output is a single bounding box (ROI) for every vehicle, coarsely marking the frontal structure of the following car. As described in Section II-C, the bounding box is temporally unstable in respect to size and location (see red boxes in Figure 3).
To smooth the bounding box locations and sizes in time, each subsequent detection at nearly the same location, covering more than 0.25% of the bounding area, is associated with the previous detection. Additionally, an exponential smoothing [22] of the box width w, height h and location u, v with a smoothing factor $\delta = 0.3$ is performed:

$$\vec{b}_{det} = [u\, v\, w\, h]^T \tag{5}$$

$$\vec{b}'_{det}(t) = \delta \cdot \vec{b}_{det}(t) + (1-\delta) \cdot \vec{b}_{det}(t-1) \tag{6}$$

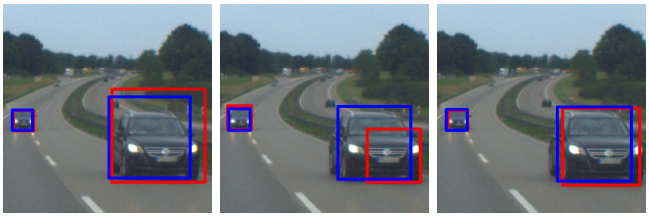

Fig. 3. Raw vehicle detection bounding boxes (red) and exponentially smoothed boxes (blue)

As a result, the unstable detections are associated with a track and smoothed in time to obtain the coarse vehicle sizes and positions in the 2D input image (see blue boxes in Figure 3). If more than two subsequent detections are associated, the track is marked as stable, and the segmentation is initialized.

SEGMENTATION For a correct estimation of the distance to the detected car, the precise ground mark of the object has to be known. Even with temporal smoothing, the lower border of the detector box is still too imprecise and noisy in time. Therefore the precise ground mark is determined by segmenting the car outline from the background on pixel level. Additionally, this segmentation is used for precise tracking and as input for the visualization pipeline (see Section IV).
The Grab Cut segmentation algorithm introduced in Section II-C requires a coarse mask of the image region in which the foreground object is located. Technically this is a mask, which defines a background area and an uncertain region, containing foreground and background pixels (bimap). Since the Grab Cut algorithm only refines the segmentation within the bounding box, it is crucial that

the box encloses the whole foreground object (vehicle). Unfortunately, the vehicle detector covers only a part of the vehicle and therefore can not be used directly as an input bimap (see Figure 3). The 2D projection of an approaching vehicle resembles a trapezoid shape, which is why the size of the bounding box is increased by 54% and morphed to a trapezoid (see red trapezoid in Figure 4). The trapezoid is used as input for the Grab Cut algorithm to perform a separate segmentation for each vehicle (see green area in Figure 4).

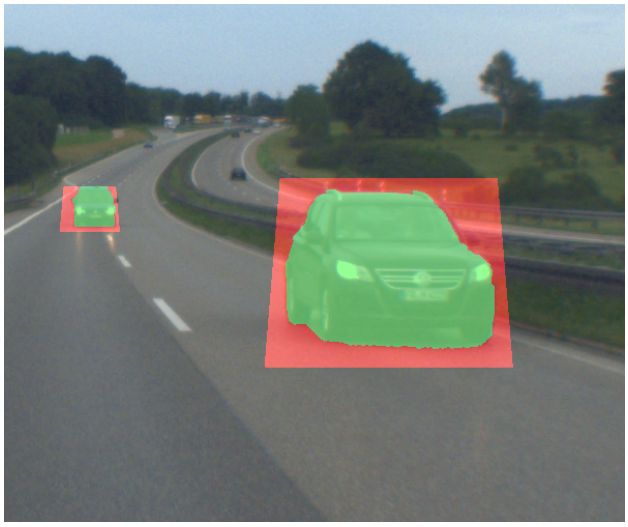

Fig. 4. Grab Cut input bimap generated out of detector results (red) and resulting segmented foreground vehicle (green)

GRABCUT TRACKING To track the vehicle in consecutive video frames, we combine the vehicle detector with the Grab Cut segmentation. To keep track of the vehicle we exploit the high sample rate of the input video (30fps): Even by taking the ego-motion and the object motion into account, the location and size of the tracked vehicle differs only slightly in consecutive frames. Hence, the new bimap is generated out of the segmented region from the previous frame, by enlarging the masked area by 30% in each direction. Afterwards the Grab Cut segmentation is performed to obtain the exact location and outline of the tracked vehicle (see Figure 4).

TRACKING VERIFICATION In some cases the Grab Cut algorithm fails either by segmenting only a small part of the vehicle or by segmenting the whole masked region as foreground. These errors can be detected by comparing the covered area of the bimap foreground and the segmentation result. A coverage below 60% or above 90% is considered invalid.

The output is corrected by a second segmentation pass with a new bimap, which is derived from the detector results of the current frame (similar to the initialization of the Grab Cut tracker). If the tracked vehicle was not detected in the current frame, the detection from the previous frame is used. If the second segmentation fails as well, the previous mask is propagated from the previous frame, taking the 2D shift $\Delta\vec{b}'_{det}(t)$ from the previous detections into account.

To validate if the current tracked object is a vehicle, the detections are associated with the active tracks. If a track is not associated with a detection in 10 consecutive frames, the track is deleted. If a new detection occurs, a new track is initialized.

DISTANCE AND VELOCITY ESTIMATION With the correct outline of the vehicle, the ground mark in image space can be assumed to be the undermost pixel of the segmentation mask. Based on the flat world assumption ($z = 0$) and the camera calibration (see Section III-A), the 3D coordinates (x_g, y_g, z_g) of the 2D ground mark (u_g, v_g) are calculated (see equation 7).

$$\begin{pmatrix} x_g \\ y_g \\ z_g \end{pmatrix} = \begin{pmatrix} \frac{z_0}{\tan\left(\arctan\left(\frac{v_g - v_0}{f_0}\right) + \theta\right)} + x_0 \\ \frac{u - u_0}{f_0} \cdot (x_g - x_0) + y_0 \\ 0 \end{pmatrix} \tag{7}$$

To estimate the relative speed $\hat{v}_t$ and time to impact ΔT_i of the vehicle, an alpha-beta-filtering of consecutive detections is performed ($\alpha = 0.5$, $\beta = 0.1$) (see equations 8 to 11).

$$\hat{x}_t = \hat{x}_{t-1} + \Delta T\, \hat{v}_{t-1} \tag{8}$$

$$\hat{x}_t = \hat{x}_t + \alpha \cdot \hat{r}_t \qquad \|\hat{r}_t = \hat{x}_t - \hat{x}_{t-1} \tag{9}$$

$$\hat{v}_t = \hat{v}_{t-1} + \frac{\beta}{\Delta T} \cdot \hat{r}_t \tag{10}$$

$$\Delta T_i = \frac{\hat{x}_t}{\hat{v}_t} \tag{11}$$

The additional information on the following vehicles are added as meta information to the track. The importance or risk potential of each vehicle is implicitly encoded in the time to impact (ΔT_i).

IV. NATURAL VISUALIZATION OF DISTANCE AND MOTION

We propose several visualization techniques to convey the gathered information and risk potential of the following vehicles to the driver. A simple visualization would be an overlay with numerical values showing the distance, speed and risk potential of each detected vehicle. With multiple following vehicles, the displayed information becomes numerous and therefore the assessment of the outgoing risk potential grows more and more difficult for the driver. We propose visualization techniques which are motivated by subconsciously processed sensory effects in the human visual system.

A. Visualize Risk Potential

User studies [23] have shown, that encoding a risk potential with signal colors alerts human observers subconsciously. In this context, red creates the strongest reaction on the human assessment of the observed scenery. This reaction gradually decreases with the colors orange through green to white. Accordingly, red is usually used to encode danger or a warning. Hence, we propose to overlay the lane of the

approaching vehicle with signal colors, to convey the risk potential of a collision in a prominent manner. Therefore the lane overlay presented in Section III-A is confined to the lane of the closest vehicle and shortened to its ground mark. The overlay color is defined by the estimated time to impact which is encoded in shades of red through orange to green (see Figure 5).

Fig. 5. Time to impact dependent color encoded lane overlay to intuitively visualize the risk potential of a lane change to the driver

B. Artificial Depth-of-Field

The proposed visualization is well suited to warn the driver in critical situations. However, for some applications it might be undesirable to manipulate the video image in such a strong manner.

To solely convey distance information in a minimally invasive way, we adopted the techniques from Roessing *et al.* [5] to render an artificial depth-of-field. The required input depth map is generated from the environment model M (see Section III) and the vehicle track meta information. Thereby a depth value is computed for all pixels and stored in a depth map $(D(u,v))$: For each pixel which was detected as road, the depth is computed in accordance to Formula 7. Afterwards, each pixel which was segmented as vehicle, gets assigned the ground mark distance value, stored in the meta data (see Section III-B). All other pixels are set to infinite distance (see Figure 6).

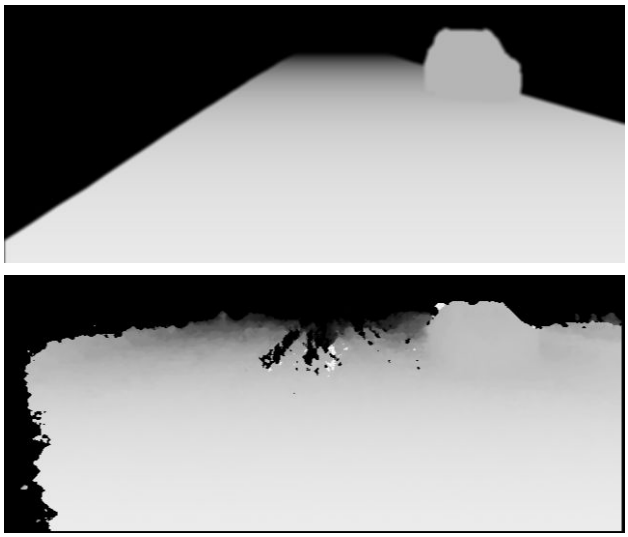

Fig. 6. Reconstructed depth map based on monocular input images (top) vs. depth map generated with a stereo rig, calculated with SGM (bottom)

The calculation of the blur radius $b(u,v)$ is based on the assumption of an overall sharp input image, the physical parameters of the human eye ($A_E = 4.6\text{mm}$, $f_E = 24\text{mm}$) and the pixel size of the camera imager ($w_p = 3.75\mu\text{m}$). The focal distance x_f is automatically set to the closest vehicle (see equation 12). The Gaussian blurring is performed as described in [5]. Pixels at infinite distance are blurred with the maximum blur radius. An artificial depth-of-field is rendered as a result, which supports the driver in focusing on the closest vehicle and conveys a coarse impression of spacial layering and distances within the presented scenery (see Figure 7).

Fig. 7. Depth map based artificial depth-of-field. To make this effect visible in print media, the blur radii are broadened by a factor of 3

$$b(u,v) = \left| A_E \frac{f_E}{x_f} \cdot \left(1 - \frac{x_f}{D(u,v)}\right) \right| \cdot \frac{1}{w_p} \tag{12}$$

C. Artificial Motion Blur

Artificial depth-of-field is able to convey coarse distance information [4]. As described in Section II-B, motion blur is able to induce a sensation of velocity. Therefore we propose an artificial motion blur rendering to convey supplemental speed information in a natural manner. The rendering is performed by calculating the 2D shift $\vec{m}_d(t)$ of the detection box center coordinates (u_d, v_d). Afterwards, the length of the motion blur vector $\vec{m}_b(t)$ is calculated by enlarging $\vec{m}_b(t)$ by a user-defined factor ($\gamma \approx 2.0$). Finally each pixel of the segmented vehicle is convoluted along the motion blur path $\vec{m}_b(t)$ with linearly decreasing intensity $I(u)$.

$$\vec{m}_d(t) = \begin{pmatrix} u_d(t) \\ v_d(t) \end{pmatrix} - \begin{pmatrix} u_d(t-1) \\ v_d(t-1) \end{pmatrix} \tag{13}$$

$$\vec{m}_b(t) = \gamma \cdot \vec{m}_d(t) \tag{14}$$

$$I(u) = 1 - \frac{|u - u_d|}{\|\vec{m}_{b,u}\|} \tag{15}$$

V. EVALUATION AND BENCHMARK

A. Reconstructed Depth Map Quality

The reconstructed depth map is based on the assumption of a flat world and plane frontal structure of the detected vehicles (see Section IV-B). This introduces an error in the reconstructed depth map. To evaluate the error, a secondary depth map was generated by a stereo camera setup with a

Fig. 8. Artificial exaggerated motion blur, supporting the driver's assessment of the approaching speed of the trailing vehicle

baseline of 25 cm incorporating the Semi Global Matching stereo algorithm [12]. The presented monocular reconstruction processes the left camera video stream of the stereo rig, to generate comparable depth maps. 60 frames from three different sequences were evaluated (see Figure 6). At each location with valid distance information in the stereo and the reconstructed depth map, the absolute mean distance error was calculated $\bar{x}_{err} = 1,95m$. This error is caused by pitching and rolling of the vehicle, violations of the flat world and plane vehicle assumption. The error and the missing distance information beside the road make this technique unsuitable for most computer vision applications. Nonetheless, the quality of the depth map is sufficient for realistic depth-of-field renderings (see Figure 7).

B. Benchmark

The presented framework was implemented in MATLAB®, interfacing a GPU implementation of the Grab Cut algorithm [21]. The mean runtime of different stages of the pipeline are averaged over three different sequences (see Table I). The longest runtime while reconstructing the 3D environment is the vehicle segmentation. The longest runtime of the visualization techniques is the rendering of the artificial depth-of-field. Blurring each pixel with an individual blur radius with large scales increases the average runtime of the depth-of-field rendering substantially.

TABLE I

AVERAGED RUNTIMES OF DIFFERENT STAGES OF THE FRAMEWORK

Processing Stage	Runtime
Vehicle Detection	0.040s
Vehicle Segmentation	0.669s
Update Environment Model	0.193s
Visualize Risk Potential	0.317s
Visualize Depth-Of-Field	53.356s
Visualize Motion Blur	1.449s

VI. USER STUDY

Roessing *et al.* performed a user study to evaluate artificial depth-of-field renderings to support the viewer in search-and-find tasks [5]. Held *et al.* proposed that artificial depth-of-field is able to induce a sensation of relative distances in 2D images [4]. Since the impact of artificial depth-of-field in human perception has already been examined, we conducted an additional user study to evaluate the impact of artificial motion cues on the sensation of velocity in video streams. Therefore, artificial motion visualizations were added to a highway rear-view video stream with the techniques described in Section IV. The main goal was to evaluate if it is possible to induce the sensation of a higher approaching speed of objects by adding artificial motion cues - even if the observed object is decelerating or driving with similar speed.

A. Study Design

Short video sequences of a following car, captured with a rear-view camera mounted on a vehicle that was driving with a constant speed on the right lane, were shown to the participants. To avoid any interference with the brand or color of the following vehicle, all sequences captured the same car (blue Mercedes B-class). To limit the influence of a conscious processing by the viewer, the length of the stimulus video was confined to 1 sec per scene. To eliminate the influence of previous knowledge, each scenery was only presented once to a single participant.

After each presented video, the user had to decide if the following car's velocity was lower, the same, higher or much higher when compared to that of the camera vehicle. If the user was uncertain in their answer, they were allowed to skip the question.

B. Study Setup

As input material 52 scenes $\big($8 slower (≈ -20km/h), 17 of the same speed ($\approx \pm 0$km/h), 25 faster ($\approx +20$km/h), 2 much faster ($\approx +40$km/h)$\big)$ were captured. Each scene, including the scenes showing a decelerating car, was rendered with artificial motion blur and the risk potential visualization. The original scene and the renderings were added to a sequence pool (53 x 3 scenes). We presented 6 or 7 randomly chosen sequences out of the pool without repetition to each of the 70 participants. To familiarize the user with the system, they were shown 4 different scenes with the 4 different velocities of the following vehicle, which were excluded from the results. Overall, 477 sequences were shown, 446 answers were valid, 31 (1 original, 26 motion blur, 4 risk potential) answers were skipped by the user. The skipped answers were excluded from the evaluation.

C. Evaluation

We performed a Pearson chi square analysis [24] to evaluate the answers. The null hypothesis was "The perceived approaching speed is not influenced by adding additional motion cues". Accordingly, the answers were classified in three categories: The perceived speed was underestimated, correctly estimated or overestimated compared to the actual speed of the following car. The first row of every category in Table II shows the observed frequency (f_o). The second row shows the expected frequency (f_e) and the quadratic residue r^2. The statistic test is performed by comparing the sum over all residues $\chi^2 = \sum r^2$ with the value for a $\chi^2(\mathrm{df}, p)$ distribution with the same degree of freedom $(\mathrm{df} = 4)$ and a high

TABLE II

CHI SQUARE ANALYSIS OF THE PERCEIVED RELATIVE SPEED OF A FOLLOWING VEHICLE. BLUE COLORED FREQUENCIES SHOW SIGNIFICANTLY LOWER SPEED ESTIMATIONS THAN THE EXPECTED VALUE. RED COLORED FREQUENCIES SHOW A SIGNIFICANTLY HIGHER ESTIMATION. ACCORDINGLY, ARTIFICIAL MOTION BLUR IS ABLE TO INCREASE THE ESTIMATED SPEED IN THE VIEWERS PERCEPTION.

Visualization	**under-estimated**	**correctly estimated**	**over-estimated**	$\sum_i$
Original	24	123	*9*	**156**
(f_e) r^2	(23) 0.01	(114) 0.70	*(19) 4.90*	
Motion Blur	17	106	28	**151**
(f_e) r^2	(23) 1.42	(110) 0.17	*(18) 5.63*	
Risk Potential	26	97	16	**139**
(f_e) r^2	(21) 1.25	(102) 0.21	(17) 0.01	
$\sum_j$	**67**	**326**	**53**	**446**

significance $(p = 0.01)$ level. The sum $\sum r^2 = 14.34$ is larger than $\chi^2(4, 0.01) = 13.2$ and therefore the null hypothesis has to be rejected. From this it follows that the visualization has an impact on the perceived speed. If $r^2 > 3.84$, the observed frequency differs significantly from the expected frequency. Accordingly, the perceived speed in the unprocessed video was not overestimated by a significant number of participants (9 out of 156), whereas the artificial motion blur led to a significant overestimation of the approaching speed (28 out of 151). The risk potential visualization showed no significance. From this one can deduce that artificial motion blur is able to induce a perception of higher speed of object motion in video streams. Thus, this technique can be used to enhance the speed perception in video streams.

VII. CONCLUSIONS AND FUTURE WORK

We presented a framework that is capable of enhancing the sensation of motion, relative speed and distance in monocular video streams to support driver's assessment of the rear traffic situation while driving on a highway.

The reconstruction of the spacial arrangement and motion vectors is solely based on monocular camera video streams, processed with odometry, lane and vehicle detection algorithms. The visible vehicles are detected within a maximum distance of $\approx 150\,\mathrm{m}$ and tracked on pixel level by incorporating the Grab Cut segmentation algorithm. The track is initialized and verified by a vehicle detector, extracting a robust track of the targeted vehicle.

As a result a supplemental depth map can be reconstructed from the monocular camera images. The distance estimations and the depth map are sufficient for depth-of-field renderings and an evaluation of the current traffic situation and risk potential. The proposed rendering techniques generate intuitive speed, risk and depth cues for the viewer. The impact of artificial motion blur on the human visual cortex and the usability of this technique to support the human assessment of the presented scenery, was evaluated in a user study.

Further work will address the realtime capabilities of this framework by speeding up the segmentation and rendering techniques with a complete GPU implementation.

Future Applications of this framework could exploit the acuity of additional sensors (e.g. Radar or Lidar), to improve the quality of the 3D reconstruction and risk potential evaluation.

REFERENCES

[1] R. Schweiger, H. Neumann, and W. Ritter, "Multiple-cue data fusion with particle filters for vehicle detection in night view automotive applications," pp. 753–758, 2005.

[2] U.-P. Kaeppeler, M. S. von Trzebiatowski, A. Gern, U. Franke, and P. Levi, "Detecting Reflection Posts - Lane Recognition on Country Roads," in *Proceedings of the 2004 IEEE Intelligent Vehiclees Symposium*. Parma: Consiglio Nazionale delle Ricerche, June 2004, Conference Paper, pp. 304–309.

[3] G. Mather, *Foundations of Sensation and Perception*. Psychology Press, 2009, vol. 2.

[4] R. T. Held, E. a. Cooper, J. F. O'Brien, and M. S. Banks, "Using blur to affect perceived distance and size," *ACM Transactions on Graphics*, vol. 29, no. 2, pp. 1–16, Mar. 2010.

[5] C. Roessing, J. Hanika, and H. P. A. Lensch, "Real-time disparity map-based pictorial depth cue enhancement." *Comput. Graph. Forum*, vol. 31, no. 2, pp. 275–284, 2012.

[6] M. Wertheimer, "Experimentelle studien ueber das sehen von bewegung," *Zeitschrift fuer Psychologie*, 1912.

[7] J. Schmid, R. W. Sumner, H. Bowles, and M. Gross, "Programmable motion effects," *ACM Transactions on Graphics*, vol. 29, no. 4, p. 1, 2010.

[8] J. Korein and N. Badler, "Temporal anti-aliasing in computer generated animation," in *Proceedings of the 10th annual conference on Computer graphics and interactive techniques*, ser. SIGGRAPH '83. New York, NY, USA: ACM, 1983, pp. 377–388.

[9] M. Potmesil and I. Chakravarty, "Modeling motion blur in computer-generated images," *SIGGRAPH Comput. Graph.*, vol. 17, no. 3, pp. 389–399, July 1983.

[10] G. Brostow and I. Essa, "Image-based motion blur for stop motion animation," *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, no. August, pp. 561–566, 2001.

[11] B. Kim and I. Essa, "Video-based nonphotorealistic and expressive illustration of motion," pp. 32–35, 2005.

[12] S. K. Gehrig and C. Rabe, "Real-Time Semi-Global Matching on the CPU," *Image Rochester NY*, pp. 1–8, 2010.

[13] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems 1," *Journal Of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[14] P. Viola and M. Jones, "Robust real-time object detection," in *Proc. of the IEEE Workshop on Statistical and Computational Theories of Vision*, 2001.

[15] M. Gabb, O. Loehlein, M. Oberlaender, and G. Heidemann, "Efficient monocular vehicle orientation estimation using a tree-based classifier," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2011.

[16] M. Gabb, R. Wagner, M. Gressmann, O. Hartmann, O. Loehlein, R. Schweiger, and K. Dietmayer, "Feature selection for automotive object detection tasks - a study," in *Proc. of the IEEE International Conference on Consumer Electronics – Berlin*, 2012.

[17] D. Comaniciu, P. Meer, and S. Member, "Mean Shift : A Robust Approach Toward Feature Space Analysis," *Analysis*, vol. 24, no. 5, pp. 603–619, 2002.

[18] M. Gressmann, O. Loehlein, and G. Palm, "Pedestrian localization," in *Prof. of the IEEE Intelligent Systems and Informatics Symposium*, 2011.

[19] A. Blake, C. Rother, M. Brown, P. Perez, and P. Torr, "Interactive Image Segmentation Using an Adaptive GMMRF Model," vol. 3021, no. 5, pp. 428–441, 2004.

[20] Y. Boykov and V. Kolmogorov, "Computing geodesics and minimal surfaces via graph cuts," *Proceedings Ninth IEEE International Conference on Computer Vision*, vol. 2003, no. Iccv, pp. 26–33, 2003.

[21] C. Rother, V. Kolmogorov, and A. Blake, ""GrabCut": interactive foreground extraction using iterated graph cuts," *Computer*, vol. 23, no. 3, pp. 309–314, 2004.

[22] P. Mertens and S. Raessler, *Prognose Rechung*. Physicas, 2004, vol. 6.

[23] W. Karwowski, *International Encyclopedia of Ergonomics and Human Factors*. Boca Raton, 2006, vol. 3.

[24] K. Pearson, "On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling," *Philosophical Magazine*, vol. 50, no. 5, pp. 157–175, 1900.