

Data Fusion Strategies in Advanced Driver Assistance Systems

2010-01-2337

Published
10/19/2010

Michael Darms, Florian Foelster, Jochen Schmidt, Dominik Froehlich and Alfred Eckert
Continental AG

Copyright © 2010 SAE International

ABSTRACT

Data fusion plays a central role in more and more automotive applications, especially for driver assistance systems. On the one hand the process of data fusion combines data and information to estimate or predict states of observed objects. On the other hand data fusion introduces abstraction layers for data description and allows building more flexible and modular systems.

The data fusion process can be divided into a low-level processing (tracking and object discrimination) and a high level processing (situation assessment). High level processing becomes more and more the focus of current research as different assistance applications will be combined into one comprehensive assistance system. Different levels/strategies for data fusion can be distinguished: Fusion on raw data level, fusion on feature level and fusion on decision level. All fusion strategies can be found in current driver assistance implementations.

The paper gives an overview of the different fusion strategies and shows their application in current driver assistance systems. For low level processing a raw data fusion approach in a stereo video system is described, as an example for feature level fusion the fusion of radar and camera data for tracking is explained. As an example for a high level fusion algorithm an approach for a situation assessment based on multiple sensors is given. The paper describes practical realizations of these examples and points out their potential to further increase traffic safety with reasonably low cost for the overall system.

INTRODUCTION

Although the volume of traffic has noticeably increased within the last 15 years improvements in both driving and

transport safety led to a cut by half of traffic fatalities in the EU. Beneath political and educational steps major improvement of active and passive vehicle safety technology led to this trend.

The European Charta for road safety [1] with the goal to cut in half the number of traffic fatalities by 2010 compared to the number of 2004 is driving forward this development. Signees of this charta support this goal in many ways. One of Continental's activities is the integral safety program ContiGuard® [2]. Many of the approaches within this integrated safety approach are based on data fusion techniques described in this paper.

Higher requirements are made on driver assistance systems that should alert the driver in case of dangerous situations or should assist him to solve these: safety related driver assistance systems. A typical representative is the application emergency brake assist [3], which integrates environment information into brake assistance. The quality and value of the information received by different single sensors and the combination through a data fusion approach determines the assistance concept and the increased safety level offered by the system.

The paper presents different data fusion strategies used in driver assistance systems which increase traffic safety. The presented approaches can be found in applications within Continentals safety activities, which show the relevance of the presented concepts in practice. In a first part a general overview over data fusion along with some basic definitions is given. The second part of the paper presents details and applications of the data fusion algorithms and shows their contribution to automotive safety applications.

DATA FUSION

According to Steinberg et al. the process of data fusion can be defined as follows: “Data fusion is the process of combining data or information to estimate or predict entity states.” [4]. Examples for entities in the context of driver assistance applications can be objects in the environment like other observed vehicles or abstract objects like the pitch angle of the ego-vehicle.

The overall goal of data fusion applications is to combine data from different single sensors in a way, which combines the strengths and reduces the weaknesses of single sensors. Typically the following aspects are addressed in data fusion setups: Redundancy of information, complementarity of information, enhancement of timing aspects, and reduction of costs (see e.g. [5]). One of the main influence factors for costs is the architecture of the fusion system [6]. Besides that the architecture design strongly influences the possibility of cooperation of partners during the development process (encapsulation/coupling of components).

The Joint Directors of Laboratories (JDL) Data Fusion Model (see e.g. [4], [7]) distinguishes low level processing and high level processing of data. Low level data processing includes track estimation and object discrimination. Track estimation stands for the estimation of states as used for control loops (e.g. position, velocity, etc.). Object discrimination can be subdivided into detection and classification. Detection refers to all actions involved to decide whether an object is present or not, classification refers to algorithms allocating an object to a predefined class (e.g. vehicle, pedestrian, etc.). High level processing includes the situation assessment algorithms and the control of the data fusion process (e.g. allocating resources).

Data Fusion can occur on different abstraction levels of information. Typically raw data fusion, feature level fusion and decision level fusion are distinguished [7]. In a raw data fusion approach minimally preprocessed data is combined. Information can be pixels of imagers for example. One major advantage of this approach is the availability of the complete information of the single sensors. This way the fusion algorithm can be optimized to use all available information. On the negative side the resulting algorithms are rather inflexible as they are specialized for a particular sensor setup. Besides, large amounts of data have to be handled in the system (e.g. transmission of data to a centralized ECU). In a feature level fusion approach features are extracted from the raw data and then the resulting information is combined. This significantly reduces the communication bandwidth between sensors at the cost of an information reduction. The system can be better modularized and more easily be extended using well defined interfaces. In a decision level approach decisions made in earlier steps of the processing chain are combined to a fused decision. This approach can be used for object

discrimination or during situation assessment. If the information underlying the decisions is independent an optimal result can be achieved while still maintaining a flexible system that can be extended with further information sources in the future. Examples of the different fusion strategies are presented in the following sections.

RAW DATA FUSION: STEREO CAMERA

In this section a stereo camera system is presented as an example of raw data fusion originating from different sensors, in this case two camera modules that are rigidly attached to each other. The raw data used here are pixels provided by two imagers. Stereo cameras become more and more interesting for safety related driver assistance systems, as they are capable of providing distance and height of arbitrary objects directly by fusion of two images. A monocular camera in contrast can provide depth from a single image only using models or specific assumptions about objects in the real world. This characteristic and the accuracy of the distance measurements for ranges up to about 70m for a typical setup in a vehicle make stereo cameras especially attractive for assistance scenarios at low speeds or when short reaction times at close range are required, e.g. in city traffic, traffic jams or for pedestrian protection. On top of stereo-based functions, the sensor system can provide the complete functionality of a monocular camera system, like traffic sign recognition or lane detection. Other sensor technologies that provide direct distance measurements like radar or lidar lack this option.

The computing steps required to generate depth from a stereo image pair are shown in [Figure 1](#). Special care has to be taken regarding image capturing: In order to generate valid depth information, both imagers have to be synchronized such that sensor elements corresponding to the same objects in the world are exposed at the same time, as moving objects will lead to inaccurate depth estimation otherwise. This is particularly important as most state-of-the-art CMOS imagers nowadays use a rolling shutter, where imager rows are exposed one after another rather than simultaneously [16].

Starting from the images, two pre-processing steps are usually required: Conversion from raw pixel data to gray values, and rectification of the stereo image pair, which facilitates depth computation.

Depending on the color filter array (CFA) used, a conversion step will be necessary to convert the raw pixels to gray values (gray conversion). This step can be omitted if no CFA is used at all. Popular choices for CFAs are Red-Clear or Bayer pattern, which allows for full color reconstruction by applying red, green, and blue filters on the sensor elements. As gray conversion depends highly on the CFA, we refer the

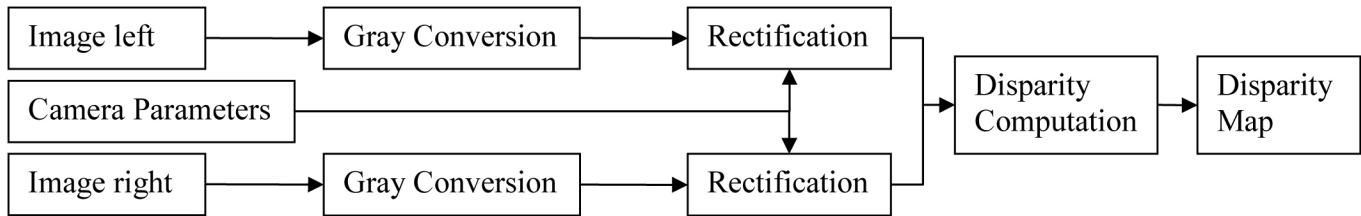


Figure 1. Stereo system overview: Raw images are converted to gray scale and rectified using intrinsic as well as extrinsic camera parameters. Rectified images are provided to the disparity computation module to facilitate correspondence search. The output is a disparity map containing depth information for each pixel.

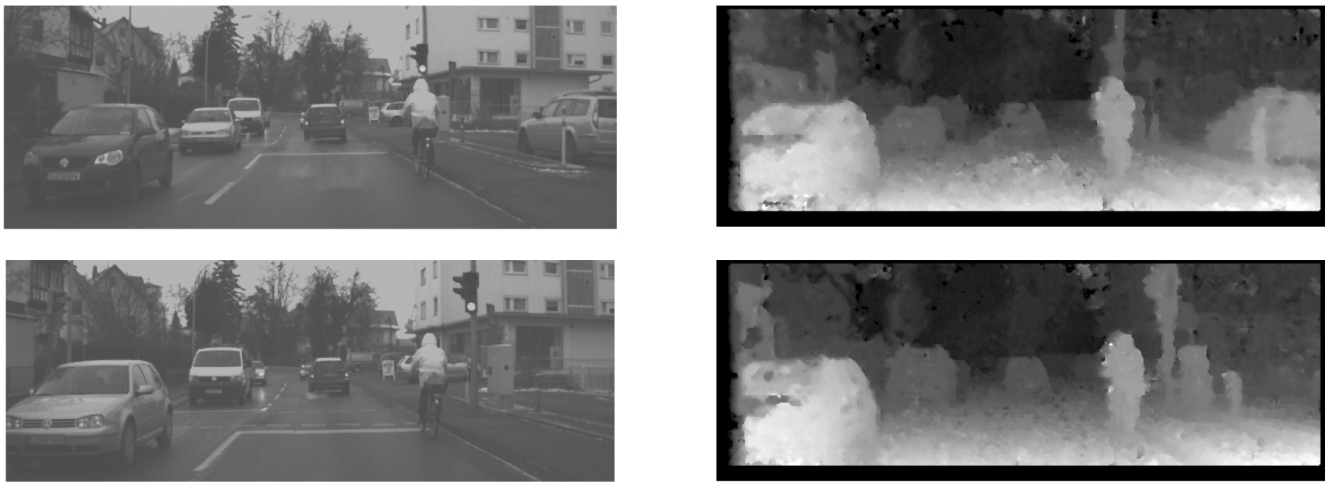


Figure 2. Disparity Map. Left: Images of traffic scenes captured in Lindau (Germany). Right: Disparity Images: The darker the pixel, the smaller the disparity (larger the distance).

reader to the literature. Details on Bayer to gray conversion can be found, e.g., in [8].

In order to compute depth information from the stereo image pair, an algorithm is required that matches object features (e.g., corners) in the left image to features in the right image, thus establishing correspondences between them. Using these correspondences and the stereo camera parameters (intrinsic as well as extrinsic ones), depth can be computed for each feature directly by triangulation (see [9] for details). This would lead to a sparse depth map of the captured scene, which is usually not sufficient for automotive applications.

Mainly, the goal of rectification is to simplify correspondence search. From computer vision it is known that stereo images are related by the epipolar constraint [9], which, amongst other things, states that corresponding points lie on epipolar lines, which all intersect in the epipole. Therefore, it is sufficient to perform correspondence search along these lines as opposed to searching the complete image. However, as these lines may have arbitrary slope, pixel interpolation is required for each search. Rectification fixes this by transforming the images in a way such that in the resulting rectified images corresponding points lie on the same

scanline. Depending on the camera parameters, rectification is normally a projective transformation [9]. The rectification step is still applied for each image separately, leading to a rectified image pair; as rectification requires known calibration parameters, there is a dependency between the two images.

Disparity map computation is the step where information from both cameras is combined. The term disparity refers to the distance between x-coordinates of corresponding points in left (x_l) and right (x_r) image measured in pixels, given by $d = x_r - x_l$. The disparity d is directly related to depth $z = f \cdot b/d$, where F is the focal length and b the baseline (distance between optical centers) of the two cameras. For each pixel in one image (or both, if required), the disparity map contains the disparity - and therefore depth - for this coordinate. An example of a disparity map with disparities coded as gray values is shown in Figure 2.

Many different algorithms for disparity map computation have been proposed; they differ in density of the resulting map, occlusion as well as sharp discontinuity handling, and robustness against illumination changes. The methods can be classified in local, global, or semi-global ones: A local