A Framework for Multiple Radar and Multiple 2D/3D Camera Fusion

Marek Schikora[1] and **Benedikt Romba**[2]
[1] FGAN-FKIE, Germany
[2] Bonn University, Germany
schikora@fgan.de, romba@uni-bonn.de

Abstract: In this paper we present a framework for the fusion of radar and image information. In the case considered here we combine information from multiple close-range radars to one fused radar measurement using the overlap region of the individual radars. This step is performed automatically using a feature based matching technique. Additionally, we use multiple 2D/3D cameras that generate (color) image and distance information. We show how to fuse these heterogeneous sensors in the context of airport runway surveillance. A possible application of this is the automatic detection of midget objects (e.g. screws) on airfields. We outline how to generate an adaptive background model for the situation on the runway from the fused sensor information. Unwanted objects on the airfield can then be detected by change detection.

1 Introduction

The simultaneous interpretation of heterogeneous sensor data, such as radar and image data, is a difficult task in information fusion. In this paper we present a framework for the fusion of these systems. The proposed algorithms are set in the context of airport runway surveillance. Every airplane starting or landing on a runway can loose some parts from its fuselage, e.g. screws. These parts can then damage the following aircrafts, which could lead to air crashes. Nowadays the airstrips are inspected visually by the airport personnel. In the following we will show how our framework can be used for the automated detection of such midget objects.

In the case consideres here we use $M \in \mathbb{N}$ close-range radars and $N \in \mathbb{N}$ 2D/3D cameras to cover the airstrip. The cameras we use are described in detail in [PHW$^+$06]. We assume the situation shown in Figure 1. In it the viewpoint change of a single camera is just a translation along the x-axis. We allow for small displacement and alignment errors.

In the following section we will outline how to perform the fusion of each sensor system. Then, in Section 3, we will demonstrate the fusion of both sensor systems. Section 4 describes briefly a background estimation strategy for the fused information, so that change detection can be performed to detect unwanted objects on the runway.

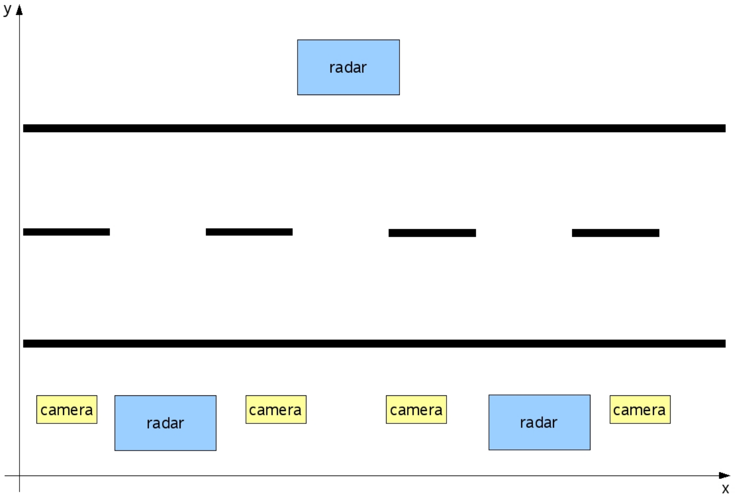


Figure 1: Situation for our fusion framework

2 Fusion of Homogeneous Information

In this section we describe how to fuse the information from a single sensor class. To solve this task we will use a feature-based approach inspired by the works of [Low04] and [MCUP02].

2.1 Radar Information

We assume that the radar data has a sufficiently high resolution, so that we can treat it like a grayscale image $I_R : \Omega \rightarrow [0, 1]$ with $\Omega \subseteq \mathbb{R}^2$, keeping in mind that each pixel carries a range and angle information. In our case we have M radar images: $I_{R_1}, I_{R_2}, ..., I_{R_M}$. The fusion task lies now in the automatic assembling of those images into a single radar image I_R containing all the single-sensor information. So we are looking for a function f_R:

$$I_R = f_R(I_{R_1}, I_{R_2}, ..., I_{R_M}) \tag{1}$$

This can be realized with a feature-based approach. In our experiments we used the SIFT (Scale Invariant Feature Transform) features proposed in [Low04]. Each selected image feature contains its local position information, in subpixel accuracy, and a 128-dimensional descriptor containing information from a window around the feature position. This descriptor can be used for matching. So in the first step we have to compute the SIFT features for all radar images. The next step is the matching, which is realized in a pairwise fashion, such that we are independent of the actual radar geometry. For two radar images

I_{R_1} and I_{R_2} we have a correspondence for a feature F_1 from image I_{R_1}, if the following equation holds:

$$\frac{\|D_{F_1} - D_{F_{21}}\|}{\|D_{F_1} - D_{F_{22}}\|} < \tau \tag{2}$$

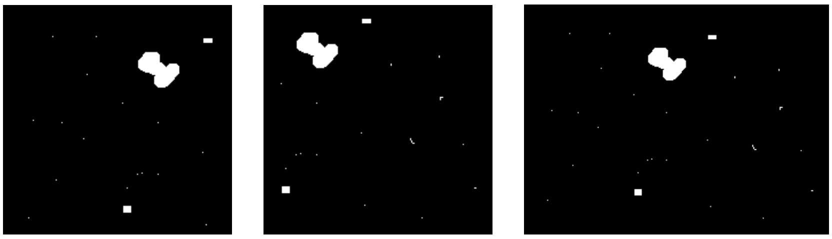
with F_{21}, from image I_{R_2}, being the nearest neighbor of the feature F_1 and F_{22}, from image I_{R_2}, being the second nearest neighbor of the feature F_1. $\tau \in \mathbb{R}$ is a threshold value. D_{F_i} denotes the descriptor of a feature F_i. A comparison of diffrent features and matching strategies can be found in [MS05]. Assuming that the point (x_i, y_i) in the first image corresponds to a point (u, v) in the second image, we can write the affine transformation between two images as:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix}. \tag{3}$$

Now we can formulate a system of linear equations using all correspondences to estimate the affine parameters a_1, a_2, a_3, a_4, t_x and t_y.

$$\begin{pmatrix} x_1 & y_1 & 0 & 0 & 1 & 0 \\ 0 & 0 & x_1 & y_1 & 0 & 1 \\ x_2 & y_2 & 0 & 0 & 1 & 0 \\ 0 & 0 & x_2 & y_2 & 0 & 1 \\ & & \cdots & & & \\ x_K & y_K & 0 & 0 & 1 & 0 \\ 0 & 0 & x_K & y_K 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ t_x \\ t_y \end{pmatrix} = \begin{pmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ \vdots \\ u_K \\ v_K \end{pmatrix} \tag{4}$$

With the solution of (4) we can construct the fused radar image I_R. Examples of this approach can be seen in Figure 2. Performance measurements of the SIFT-algorithm can be found in [MS05] and [BETG08].



a) b) c)

Figure 2: Fusion of two radar images. a) and b) simulated radar images with noise and clutter. c) Fused radar image I_R.

2.2 2D/3D Image Information

The 2D/3D PMD-camera produces two kinds of images, a conventional color Image $I_C : \Omega \rightarrow\ [0,1] \times [0,1] \times [0,1]$ and a distance image $I_D : \Omega \rightarrow \mathbb{R}_+$. Since both images are taken by the same camera a fusion of those images is already done by the hardware. The fusion of multiple 2D/3D cameras can be realized analog to Section 2.1. Examples of this matching can be found in Figure 3.

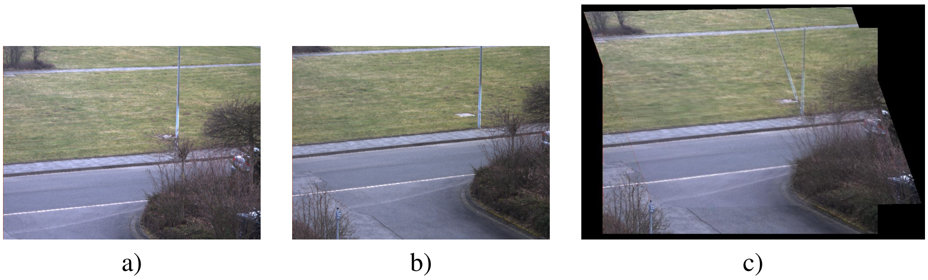


Figure 3: Fusion of two camera images I_{C_1}, I_{C_2}. a) and b) input images . c) Fused image I_C.

3 Fusion of Radar and 2D/3D Image Information

In this section we describe the fusion of radar and image data. For this purpose we need to know the scene geometry in contrast to Section 2. Each camera geometry is defined by six parameters. The location of camera $n = 1, 2, .., N$ will be denoted as $\mathbf{X}_n$ and the rotation parameters will be written in a 3×3 rotation matrix $\mathbf{R}_n$. Additionally, we need to know the intrinsic camera parameters. This information is encoded in the 3×3 matrix $\mathbf{K}_n$. With this we can compute for each pixel $(u, v) \in \Omega$ in image I_n taken by camera n its 3D coordinates. For this we write the the pixel (u, v) in homogeneous coordinates as $\mathbf{u} = (u, v, 1)^T$. The projection matrix [HZ04] of camera n can be written as:

$$\mathbf{P}_n = \mathbf{K}_n \mathbf{R}_n \left[\mathbf{I} | - \mathbf{X}_n \right] \tag{5}$$

with $\mathbf{I}$ the identity matrix. With this we can write

$$\begin{aligned} \lambda \cdot \mathbf{u} &= \mathbf{P}_n \cdot \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{A} | \mathbf{a} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} & (6) \\ \mathbf{X} &= \mathbf{A}^{-1} \left(\lambda \mathbf{u} - \mathbf{a} \right), & (7) \end{aligned}$$

where λ is the distance between the camera and the object in the real world taken from the corresponding distance image I_{D_n}. $\mathbf{X}$ is a vector containing the desired cartesian coordinates on the airfield. In Section 2 we have mentioned that each pixel of a radar image also encodes the distance and angle to the radar sensor. With this we can compute the corresponding cartesian coordinates since the position of the radar sensors are known. This can be done with the following equation:

$$
\begin{aligned}
x &= X_R - \left(\frac{s_x}{2} + x'\right) \cdot r && (8)\\
y &= Y_R + (s_y - y') \cdot r && (9)\\
z &= 0 && (10)
\end{aligned}
$$

Here x, y, z denote the cartesian coordinates, r the resolution of the radar system, s_x, s_y the dimensions of the radar image I_R, x', y' are pixel coordinates in I_R, and X_R, Y_R are the cartesian coordinates of the radar sensor. The height is set to zero since the runway is a plane. Now we know for each position on the runway the corresponding camera and radar pixel.

4 Background Estimation

As a possible application we will now briefly describe how such fused information can be used the estimate a backround model. We divide the runway into a sensor coverage grid, so that we can say for each point on this grid which camera and which radar sensor can provide information for it. This defines a five-channel image $S : \Omega \to \mathbb{R}^5$. Three channels are occupied by the visual information, one is used for the distance information and one for the radar information. Each channel contains values in the range $[0, 1]$. Now we will shortly outline a background estimation scheme for a such an image. The presented method is based on [EHD00] and [SG99]. We extend these methods to five-dimensional images and introduce in this context an independent standard deviation for each channel.

The basic idea is the following: Represent the background for a single point in this grid for each channel independently using an adaptive mixture of Gaussian models. For each channel of a point $\mathbf{x}$ we can interpret this mixture of models as a probability density function quantifying the likelihood that a value belongs to the background. Given a new situation image S_t at the time step t, we first check for a point $\mathbf{x}$ if it is foreground our background. This can be done using the value of $S_t(\mathbf{x})$ and comparing it with the background model of time step $t - 1$. If a point $\mathbf{x}$ does not belong to the foreground, then we use its current information to update the background model for the timestep t. For the sake of conciseness we will only describe this step for a single channel c. Let $S_{c,t}(\mathbf{x})$ be the value of the channel c from $S_t(\mathbf{x})$.

This update is split into two stages: initialization stage and operating stage. For a better understanding we will begin with the derivation of the operation stage and then the initialization stage.

First we check if $S_{c,t}(\mathbf{x})$ belongs to one of the already existing Gaussian models (e.g. model number j). If this is the case, then we perform the following updates:

$$
\mu_{j,t} = (1 - \rho)\mu_{j,t-1} + \rho S_{c,t}(\mathbf{x}) \qquad (11)
$$

$$\begin{aligned}\sigma_{j,t}^2 &= (1-\rho)\sigma_{j,t-1}^2 + \rho\left(S_{c,t}(\mathbf{x}) - \mu_{j,t}\right)^2 & (12)\\ w_{j,t} &= (1-\alpha)w_{j,t-1} + \alpha, & (13)\end{aligned}$$

with

$$\rho = \alpha \cdot L\left(S_{c,t}(\mathbf{x})|\mu_{j,t}, \sigma_{j,t}^2\right). \qquad (14)$$

Here, $\mu_{j,t}$ and $\sigma_{j,t}^2$ are the mean and variance of model j at time t. $\alpha \in (0,1)$ is a learning parameter that controls how fast the model adapts to a new situation. $L\left(S_{c,t}(\mathbf{x})|\mu_{j,t}, \sigma_{j,t}^2\right)$ is the likelihood for the measurement $S_{c,t}(\mathbf{x})$, given a model j. $w_{j,t}$ is the weight of model j at the time t. If these steps are preformed for model j, we have to adjust the weights of all other models $i = 1, ..., J$ with $i \neq j$:

$$w_{i,t} = (1-\alpha)w_{i,t-1} \qquad (15)$$

and to normalize the weights.

Given the case that $S_{c,t}(\mathbf{x})$ does not belong to any model, we create a new model with $S_{c,t}(\mathbf{x})$ as mean value, a large standard deviation and a small weight. If the number models exceeds a certain maximum number of models we discard, before creating a new model, the model with the smallest weight.

In the first stage we have to initialize our Gaussian models. This works similar to the already described operating stage. It is important for a correct generation process, that no unwanted object lies on the airstrip.This process could be simply done by collecting the data over some timesteps and then calculating mean and variance. Because of the great amout of data in our problem, this would cost of a lot of storage, so we only want to save the data of the actual timestep. The mean at the timestep t can be exact calculated, while the calculation of the variance is an approximation that converges step-by-step to its true value. We work with the equations (11), (12), (13), (15) but use $\rho = 1/t$. The initialphase ends after a given number of timesteps.

With this background model for all channels we can detect for each channel separately if a point belongs to the foreground. Since we are using heterogeneous channels with different characteristics, we can combine this information in a intelligent fusion system, so we will be able to detect midget objects safely.

5 Conclusion

In this paper we outlined a framework for the fusion of radar and 2D/3D image data. We presented an algorithm with the application of airport runway surveillance for the detection of unwanted midget objects on the airstrip in mind. For this task we first demonstrated how to perform a fusion of multiple radar sensors and showed that this concept can also be used for 2D/3D image data. In the next step we were able to show how to combine the radar and the camera information. As a first application we briefly described a backround estimation method for this case.

In future work we will focus on the intelligent object detection given such a background model. Additionally, we will prove our simulation results with real radar data.

References

[BETG08] H. Bay, A. Ess, T. Tuytelaars, and L.v. Gool. "Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding (CVIU)*, 110(3): pp. 246–259, 2008.

[EHD00] Ahmed M. Elgammal, David Harwood, and Larry S. Davis. Non-parametric Model for Background Subtraction. In *ECCV '00: Proceedings of the 6th European Conference on Computer Vision-Part II*, pages 751–767, London, UK, 2000. Springer-Verlag.

[HZ04] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[Low04] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[MCUP02] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. In *British Machine Vision Conference (BMVC)*, pages 384–393, 2002.

[MS05] K. Mikolajczyk and Cordelia Schmid. A Performance Evaluation of Local Descriptors. 27(10), October 2005.

[PHW$^+$06] Prasad, Hartmann, Weihs, Ghobadi, and Sluiter. Frist steps in enhancing 3D vision technique using 2D/3D sensors. In *Computer Vision Winter Workshop*, 2006.

[SG99] C. Stauffer and W.E.L. Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition CVPR*, 1999.