Intelligent Autonomous Primary 3D Feature Extraction in Vehicle System Dynamics' Analysis: Theory and Application

Annamária R. Várkonyi-Kóczy

Dept. of Measurement and Information Systems
Budapest University of Technology and Economics
Integrated Intelligent Systems Japanese-Hungarian Laboratory
Magyar tudósok krt. 2, H-1521 Budapest, Hungary
e-mail: koczy@mit.bme.hu

Abstract: 3D model reconstruction plays a very important role in computer vision as well as in different engineering applications. The determination of the 3D model from multiple images is of key importance. One of the most important difficulties in autonomous 3D reconstruction is the (automatic) selection of the 'significant' points which carry information about the shape of the 3D bodies i.e. are characteristic from the model point of view. Another problem to be solved is the point correspondence matching in different images.

In this paper a 3D reconstruction technique is introduced, which is capable to determine the 3D model of a scene without any external (human) intervention. The method is based on recent results of image processing, epipolar geometry, and intelligent and soft techniques.

Possible applications of the presented algorithm in vehicle system dynamics are also presented. The results can be applied advantageously at other engineering fields, like car-crash analysis, robot guiding, object recognition, supervision of 3D scenes, etc,. as well.

Keywords: 3D reconstruction, perspective geometry, point correspondence matching, epipolar geometry, fuzzy image processing, features extraction, information enhancement, image understanding, car-body deformation modeling, crash analysis

1 Introduction

3D reconstruction from images is a common issue of several research domains. In recent time the interest in 3D models has dramatically increased [1] [2]. More and more applications are using computer generated models. The main difficulty lies with the model acquisition. Although, more tools are at hand to ease the generation of models, it is still a time consuming and expensive process. In many cases models of existing scenes or objects are desired. Traditional solutions

include the use of stereo rigs, laser range scanners, and other 3D digitizing devices. These devices are often very expensive and require careful handling, and complex calibration procedures.

Creating photorealistic 3D models of a scene from multiple photographs is a fundamental problem in computer vision and in image based modeling. The emphasis for most computer vision algorithms is on automatic reconstruction of the scene with little or no user interaction [3].

In this paper an alternative approach is proposed which avoids most of the problems mentioned above. The object which has to be modeled is recorded from different viewpoints by a camera. The relative position and orientation of the camera and its calibration parameters will automatically be retrieved from image data. For the reconstruction we use characteristic features, like edges and corner points of the objects. The complexity of the technique is kept low on one hand by filtering out the points and edges carrying non-primary information (i.e. the so-called texture edges and points) while on the other by applying recent methods of digital image processing (see e.g. [4]-[7]) combined with intelligent and soft (e.g. fuzzy) techniques. This makes possible e.g. autonomous point correspondence matching which is the hardest step in 3D reconstruction and the biggest difficulty towards the automation of the procedure.

The introduced autonomous 3D reconstruction and its algorithms can be applied advantageously at many fields of engineering. In the second part of this paper, we will show a possible application in vehicle system dynamics: the usage in car-crash analysis.

Crash and catastrophe analysis has been a rather seldom discussed field of traditional engineering in the past. In recent time, both the research and theoretical analyses have become the part of the everyday planning work (see e.g. [8]). The most interesting point in crash analysis is that even though the crash situations are random probability variables, the deterministic view plays an important role in them. The stochastic view, statistical analysis, and frequency testing all concern past accidents. Crash situations, which occur the most frequently (e.g. the characteristic features of the crash partner, the direction of the impact, the before-crash speed, etc.) are chosen from these statistics and are used as initial parameters of crash tests. These tests are quite expensive, thus only some hundred tests per factory are realized annually, which is not a sufficient amount for accident safety. For the construction of optimal car-body structures, more crash-tests were needed. Therefore, real-life tests are supplemented by computer-based simulations, which increases the number of analyzed cases to 1-2 thousands. The computer-based simulations – like the tests – are limited to precisely defined deterministic cases. The statistics are used for the strategy planning of the analysis. The above mentioned example clearly shows that the stochastic view doesn't exclude the deterministic methods.

Crash analysis is very helpful for experts of road vehicle accidents, as well, since their work requires simulations and data, which are as close to the reality as possible. By introducing intelligent methods and algorithms, we can make the simulations more precise and so contribute towards the determination of the factors causing the accident.

The results of the analysis of crashed cars, among which the energy absorbed by the deformed car-body is one of the most important, are of significance at other fields, as well. They also carry information about the deformation process itself and may have a direct effect on the safety of the persons sitting in the car. Thus, through the analysis of traffic accidents and crash tests we can obtain information concerning the vehicle which can be of help in modifying the structure/parameters to improve its future safety. The ever-increasing need for more correct techniques, which use less computational time and can widely be applied results in the demand and acceptance of new modeling and calculating methods.

The techniques of deformation energy estimation used up till now can be classified into two main groups: The first one applies the method of finite elements [9]. This procedure is enough accurate and is suitable for simulating the deformation process, but this kind of simulation requires very detailed knowledge about the parameters of the car-body and its energy absorbing properties, which in most of the cases are not available. Furthermore, if we want to get enough accurate results, its complexity can be very high.

The other group covers the so called energy grid based methods, which starts from known crash test data and from the shape of the deformation or from the maximum car-body deformation [10]. The distribution of the energy, which can be absorbed by the cells, is considered just in 2D and the shape of the deformation is described also by a 2D curve which equals the border of the deformation visible from the top view of the car-body. The accuracy of this technique is not acceptable: In many of the cases, the shape of deformation can not be described in 2D and furthermore, the energy absorbing properties of the car-body change along the vertical axis as well causing serious impreciseness in the results.

In this paper new methods are introduced which avoid the above discussed disadvantages of the recently used techniques. First, the energy distribution is considered in 3D. Secondly, for the description of the shape of deformation spline surfaces are used, which are very suitable for modeling complex deformation surfaces. Third, the computational time and cost need is significantly decreased while the accuracy is increased by the application of intelligent techniques. Last, the deformation surface is obtained by a new 3D reconstruction method using only digital photos of the crashed car-body as input.

The methods presented in this paper can be applied at different fields of engineering. In this paper, we will show how can we construct a system capable to automatically build the 3D model of a crashed car as well as to determine the energy absorbed by the car-body deformation and the speed of the crash.

The paper is organized as follows: In Section II the primary edge extraction method is summarized. Section III is devoted to the 3D model estimation from multiple images, while Section IV is devoted to the conclusions. Intelligent applications in vehicle system dynamics and examples of the presented methods can be followed in the second part of the paper.

2 Primary Edge Extraction

Images usually contain a lot of different edges, among which there are texture edges and object contour edges, as well. From the point of view of scene reconstruction and image retrieval the latter ones are important because they carry the primary information about the shape of the objects. In we considered all of the possible edges during the model building/ searching/comparison, it would cause that the complexity/ time need of the procedure might grow to a possibly intolerable degree and furthermore, the (probably high number of) non-important details (edges) might lead to false decisions and increased the uncertainty of the modeling or caused that we disregarded recognizing an object. As a consequence, the separation of the 'significant' and 'unimportant' subsets of the edges, i.e. the enhancement of those ones which correspond to the object boundaries and thus carry primary information and the filtering out of the others which represent information of minor importance, not only significantly decreases the computational complexity of the processing but is of key importance from interpretation point of view.

2.1 Surface Smoothing

Let S_t be the surface describing the image to be processed, i.e. $S_t = \{(x, y, z); z = I(x,y,t)\}$, where the variables x and y represent the horizontal and vertical coordinates of the pixel, z stands for the luminance value, which is the function of the pixel coordinates and of the time t. The smoothing is performed by image surface deformation. Such a process preserves the main edges (contours) in the image. The surface deformation process satisfies the following differential equation [11]:

$$\frac{\partial I_t}{\partial t} = k\mathbf{n}, \tag{1}$$

where k corresponds to the 'speed' of the deformation along the normal direction **n**. In our case, this value k is represented by the mean curvature of the surface at location $[x, y]$, i.e. the speed of the deformation at a point will be the function of the mean curvature at that point. The mean curvature is defined as:

$$k = \frac{k_1 + k_2}{2}, \tag{2}$$

where k_1 and k_2 stand for the principal curvatures. Starting from equation (2), the following partial differential equation can be derived (for details, see [12]):

$$k = \frac{(1+I_y^2)I_{xx} - 2I_x I_y I_{xy} + (1+I_x^2)I_{yy}}{2\left(1+I_x^2+I_y^2\right)^{3/2}}. \tag{3}$$

Here I_x, I_y, I_{xx}, I_{xy}, I_{yy} stand for the partial derivatives with respect to the variables indicated as lower indices. Starting from equation (1) the surface at time $t+\Delta t$ (for small Δt) can be calculated as follows [11]:

$$\begin{aligned} &I(x,y,t+\Delta t) = \\ &\quad I(x,y,t) + k\sqrt{1+I_x^2(x,y,t)+I_y^2(x,y,t)}\Delta t + o(\Delta t) \end{aligned} \tag{4}$$

where $o(\Delta t)$ represents the error of the approximation.

2.2 Edge Detection

Let $z_{0,x,y}$ be the pixel luminance at location $[x,y]$ in the original image. Let us consider the group of neighboring pixels which belong to a 3x3 window centered on $z_{0,x,y}$.

The output of the edge detector is yielded by the following equation:

$$\begin{aligned} z^p_{x,y} &= (L-1)MAX\left\{m_{LA}(\Delta v_1), m_{LA}(\Delta v_2)\right\} \\ \Delta v_1 &= \left|z_{0,x-1,y} - z_{0,x,y}\right| \\ \Delta v_2 &= \left|z_{0,x,y-1} - z_{0,x,y}\right| \end{aligned} \tag{5}$$

where $z^p_{x,y}$ denotes the pixel luminance in the edge detected image and m_{LA} stands for the used membership function (see Figure 1). $z_{0,x-1,y}$ and $z_{0,x,y-1}$ correspond to the luminance values of the left and upper neighbors of the processed pixel at location $[x,y]$. L-1 equals to the maximum luminence value (e.g. 255). For more details see [5].

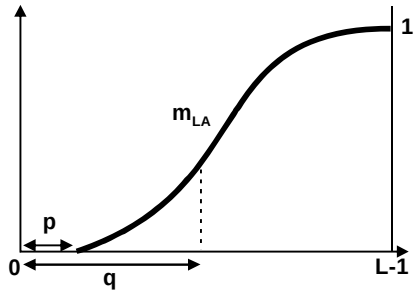


Figure 1
Fuzzy membership function m_{LA} of 'edge'. L-1 equals to the maximum intensity value, p and q are tuning parameters

2.3 Edge Separation

The obtained smoothed image is used for extracting the most characteristic edges of the objects. The procedure is performed as follows:

For each edge point taken from the edge map of the original image, the environment of the point is analyzed in the smoothed image. The analysis is realized by calculating the mean squared deviation of the color components (in case of grayscale images the gray-level component) in the environment of the selected edge point.

Let $\mathbf{p}=[p_x, p_y]$ be an edge point in the original image and let **M** denote a rectangular environment of **p** with width w and height h. The mean squared deviation is calculated as follows:

$$d = \frac{\sum_{i=p_x-w/2}^{p_x+w/2} \sum_{j=p_y-h/2}^{p_y+h/2} (\mu - I(i,j,t_{stop}))^2}{hw}, \tag{6}$$

where t_{stop} represents the duration of the surface deformation. In case of grayscale images, μ denotes the average gray level inside the environment **M**. For color images, the whole process should be done for each component separately and in this case μ corresponds to the average level of this color component inside the environment **M**.

If the so calculated deviation exceeds a predefined threshold value, then the edge point is considered as useful edge. As result, an image containing only the most characteristic edges is obtained.

3 3D Model Estimation from Multiple Images

The topic of building 3D models from images is a relatively new research area in computer vision and, especially when the objects are irregular, not finished at all. In the field of computer vision, the main work is done at one hand on the automation of the reconstruction while on the other on the implementation of an intelligent human-like system, which is capable to extract relevant information from image data and not by all means on building a detailed and accurate 3D model like usually in photogrammetry is. For this purpose, i.e. to get the 3D model of scenes, to limit/delimit the objects in the picture from each other is a key importance [13].

3.1 Noise Smoothing

As the first step, the pictures, used in the 3D object reconstruction are preprocessed. As a result of the preprocessing procedure the noise is eliminated. For this purpose we use a special fuzzy system characterized by an IF-THEN-ELSE structure and a specific inference mechanism proposed by Russo [4], [6]. Different noise statistics can be addressed by adopting different combinations of fuzzy sets and rules.

Let $I(\mathbf{r})$ be the pixel luminance at location $\mathbf{r}=[x,y]$ in the noisy image, where x is the horizontal and y the vertical coordinate of the pixel. Let $I_0 = I(\mathbf{r}_0)$ denote the luminance of the input sample having position $\mathbf{r}_0$ and being smoothed by a fuzzy filter. The input variables of the fuzzy filter are the amplitude differences defined by:

$$\Delta I_j = I_j - I_0, \; j = 1,...,8 \tag{7}$$

where the $I_j = I(\mathbf{r}_j)$, $j=1,\ldots,8$ values are the luminance values of the neighboring pixels of the actually processed pixel $\mathbf{r}_0$ (see Figure 2a). Let K_0 be the luminance of the pixel having the same position as $\mathbf{r}_0$ in the output image. This value is determined by the following relationship:

$$K_0 = I_0 + \Delta I \tag{8}$$

where ΔI is determined later (see (11)).

Let $W = \bigcup_{i=1}^{9} W_i$ be defined by a subset of the eight neighboring pixels around $\mathbf{r}_0$. Let the rule base deal with the pixel patterns $W_1,\ldots,W_9$ (see Figure 2b). The value K_0 can be calculated, as follows:

$$\lambda = MAX\left\{MIN\left\{m_{LP}(\Delta I_j) : r_j \in W_i\right\}; i = 1,...,9\right\} \tag{9}$$

$$\lambda^* = MAX\left\{MIN\left\{m_{LN}(\Delta I_j) : r_j \in W_i\right\}; i = 1,...,9\right\} \tag{10}$$

$$\Delta I = (L-1)\Delta\lambda$$
$$K_0 = I_0 + \Delta I \tag{11}$$

where $\Delta\lambda=\lambda-\lambda^*$, L is the maximum of the gray level intensity, m_{LP} and m_{LN} correspond to the membership functions and $m_{LP}(I)=m_{LN}(-I)$ (see Figure 2c). The filter is recursively applied to the input data.

r_1	r_2	r_3
r_4	r_0	r_5
r_6	r_7	r_8

Figure 2a
The neighboring pixels of the actually processed pixel r_0

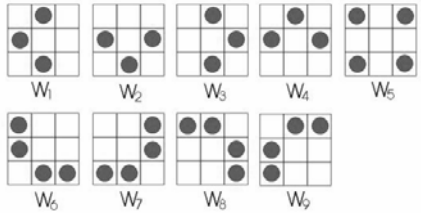


Figure 2b
Pixel Patterns

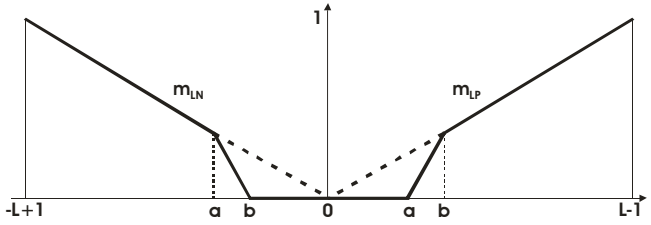


Figure 2c
Membership functions m_{LN} (large negative) and $m_{LP,}$ (large positive), a and b are parameters for the tuning of the sensitivity to noise of the filtering

3.2 Corner Detection

The edge and corner points are the most characteristic feature points in an image. For our modeling system the determination of the corners are very important. The applied corner detection algorithm utilizes the principles of the fuzzy filters and

edge detection algorithms of Russo. Besides fuzzy reasoning it uses a local structure matrix composed of the partial derivatives of the gray level intensity of the pixels. As input, we consider the noiseless and smoothed image, while as output the corners are got. A corner is indicated by two strong edges [14].

Most of the corner detection algorithms are derived from a so called local structure matrix, which has the form of

$$\mathbf{L}(x,y) = G(x,y) * \begin{bmatrix} \left(\frac{\partial I}{\partial x}\right)^2 & \left(\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}\right) \\ \left(\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}\right) & \left(\frac{\partial I}{\partial y}\right)^2 \end{bmatrix}, \tag{13}$$

where $G(x,y)$ represents the so called 2D Gaussian hump and * stands for the convolution. One of the corner detection algorithms, which uses the above local structure matrix is the Förstner's one. Förstner determines corners as local maxima of function $H(x,y)$ [15].

$$H(x,y) = \frac{\left(\frac{\partial I}{\partial x}\right)^2\left(\frac{\partial I}{\partial y}\right)^2 - \left(\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}\right)^2}{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2} \tag{14}$$

In most of the cases we can not unambiguously determine that the analyzed image point is a corner or not with only the help of a certain concrete threshold value. Therefore in the proposed algorithm, fuzzy techniques are applied for the calculation of the values (corners) which increases the rate of correct corner detection. As higher the calculated H value as higher the membership value, that the analyzed pixel is a corner. Fuzzifying the H values into fuzzy sets and applying a fuzzy rulebase we can evaluate the 'cornerness' of an analyzed pixel. This property of the pixel can advantageously be used also at the searching for the corresponding corner points in stereo image pairs (point correspondence matching), which is an indefinite step of the automatic 3D reconstruction (see also [16] and [17]).

3.3 Point Correspondence Matching and Determination of the 3D Coordinates of the Corner Points

For increasing the efficiency of the process, before starting with the actual model building, we can filter out the non-significant (texture type) edges and corners of the pre-processed images. The next step is the determination of the 3D coordinates of the remaining, primary edge points of the object. First the corner point correspondences are determined which is followed by the determination of the

edge correspondences. If the angle between the camera positions of the different images is relatively small then after the evaluation of the projection matrices of the images the corresponding points can be calculated automatically with high reliability in each image. The problem to overcome is that a point determines not another point but a line (the so called epipolar line) in the other images. To decrease the number of candidate points, first, we search the corner or edge points of the epipolar line, i.e. those points which belong to a corner or edge of the image and then the fuzzy measure of the differences of the environment of the points are minimized. The similarity of the above mentioned 'cornerness' is also considered. Having the most probable point correspondences we can calculate the 3D position of the image points and in the knowledge of the 3D coordinates of the significant points the spatial model of the object can easily be built.

3.4 Epipolar Geometry

Epipolar geometry exists between a two camera system. An important practical application of epipolar geometry is to aid the search for corresponding points, reducing it from the entire second image to a single epipolar line. The epipolar geometry can easily be found from a few point correspondences. Consider the case of two perspective images of a scene illustrated by Figure 3. The 3D point **M** is projected to point $\mathbf{m_1}$ in the left image and $\mathbf{m_2}$ in the right one. Let $\mathbf{C_1}$ and $\mathbf{C_2}$ be the centers of projection of the left and right cameras, respectively. Points $\mathbf{m_1}$ in the first image and $\mathbf{m_2}$ in the second image are the imaged points of the point **M** of the 3D space. The epipolar constraint can be written as

$$\mathbf{m}_2^T \mathbf{F} \mathbf{m}_1 = 0 . \tag{15}$$

F is known as the fundamental matrix, which defines a bilinear constraint between the coordinates of the corresponding image points. If $\mathbf{m_2}$ is the point in the second image corresponding to $\mathbf{m_1}$, it must lie on the epipolar line $\mathbf{l_{m1}}$ (see Figure 3).

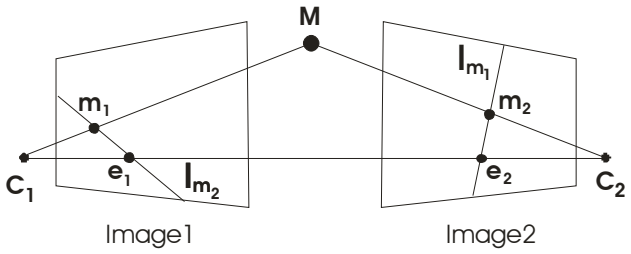


Figure 3

Illustration of epipolar geometry ($\mathbf{e_1}$ and $\mathbf{e_2}$ are the epipoles, $\mathbf{m_1}$ and $\mathbf{m_2}$ the corresponding image points, $\mathbf{l_{m1}}$ and $\mathbf{l_{m2}}$ are the epipolar lines, $\mathbf{C_1}$ and $\mathbf{C_2}$ are the camera positions. **M** is the projected 3D point)

3.5 Image Point Matching

First we have to find the most characteristic image points. These points are the corners of the analyzed object. Corners can effectively be detected with the help of the fuzzy based corner detector. Then, for each detected corner we have to determine the corresponding epipolar line. We know that the corresponding point will lay (in fuzzy sense) on this epipolar line and is also a corner point (see Fig. 4). Thereinafter the fuzzy measure of the differences of the environments of the point to be matched and the so got candidate points are minimized by a fuzzy based searching algorithm (see Figs. 4 and 5) which determines the most probably corresponding point. The same procedure is applied to edge points. For details see [13], [16], and [18].

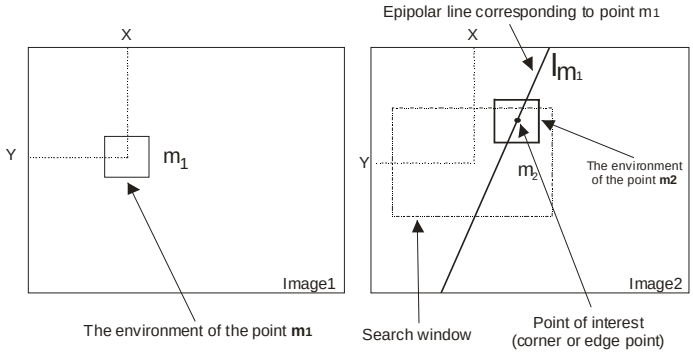


Figure 4
Illustration of the matching algorithm

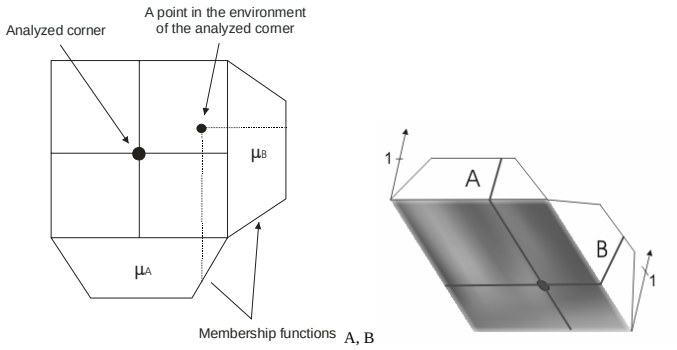


Figure 5
Illustration of an image point from the environment of the point $\mathbf{m_2}$ (see Figure 4) and the corresponding values of the membership functions of the fuzzy sets A and B)

3.6 Camera Calibration by Estimation of the Perspective Projection Matrix

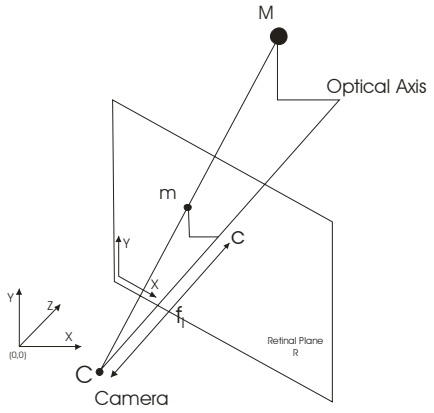


Figure 6
Perspective projection – illustration of points **M**=[*X,Y,Z*] and its projection **m**=[*x,y*] in the retinal plane *R*

There exists a collineation, which maps the projective space to the camera's retinal plane: 3D to 2D. Then the coordinates of a 3D point $\mathbf{M} = [M_X, M_Y, M_Z]^T$ (determined in an Euclidean world coordinate system) and the retinal image coordinates $\mathbf{m} = [m_x, m_y]^T$ (see Figure 6) are related by the following equations:

$$W\mathbf{m} = \mathbf{PM} \tag{16}$$

$$\begin{bmatrix} m_x W \\ m_y W \\ W \end{bmatrix} = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & 1 \end{bmatrix} \begin{bmatrix} M_X \\ M_Y \\ M_Z \\ 1 \end{bmatrix} \tag{17}$$

where W is a scale factor, $\mathbf{m} = [m_x, m_y, 1]^T$ and $\mathbf{M} = [M_X, M_Y, M_Z, 1]^T$ are the homogeneous coordinates of points **m** and **M**, and **P** is a 3 × 4 matrix representing the collineation 3D to 2D. One parameter of **P** can be fixed ($l = 1$). **P** is called perspective projection matrix. Values *a, b, c, d, e, f, g, h, i, j, k* are the elements of the projection matrix **P**. It is clear that

$$W = iM_X + jM_Y + kM_Z + 1 \tag{18}$$

From (17) we can calculate the coordinates of point **m** (m_x, m_y), as follows:

$$m_x = \frac{aM_X + bM_Y + cM_Z + d}{W} \tag{19}$$

$$m_y = \frac{eM_X + fM_Y + gM_Z + h}{W} \tag{20}$$

All together we have eleven unknowns (the elements of the projection matrix) that means that we need six points to determine the projection matrix. For more details see [19].

4 Primary Edge Extraction

The basic concept of the primary edge extraction method described in the first part includes the following steps: Consider that we have an image and we want to extract the edges corresponding to the object contours.

As the first step, it is necessary to remove the unimportant details from the image. The smoothing procedure used for this purpose is based on surface deformation. After smoothing the image, only the most characteristic contours are kept.

Next, the edge map of the original image is constructed using the fuzzy-based edge detection method described in [24]. Such an edge map contains all the possible edges.

After this step, the two processed images – the smoothed one and the edge map of the original image – are analyzed simultaneously in the following way: In case of each of the edge points a small environment of the point is taken in the original image and using the smoothed image the variance of the color components inside of this environment is analyzed. If the variance is below a predefined threshold value then the edge point is removed while otherwise it is considered as a useful, primary edge point.

The effectivity of the above information enhancement method detailed in Section II of the first part of this paper is illustrated by two simple examples. In all of the examples color images are used. The figures allow the comparison of the original edge map and the edges after applying the proposed method.

Figs. 7 and 8 are illustrations for the virtual process of changing of an image surface along the time. The two examples are fine fragments of the next example (surface of the car) at time 0 and at time t_{stop}, respectively.

The next example analyses a photo taken of a car. In Figure 9 the original image can be seen, while Figure 10 shows the smoothed image using the discussed surface deformation. Figs. 11 and 12 represent the edge maps before and after the processing, respectively. As you can see in Figure 12 many of the details disappear after the processing and only the charac-teristic edges of the car are left. This helps filtering out the non-important details and enhancing the most significant features/objects in images thus making easier image retrieval, object recognition, reconstruction of scenes, etc.

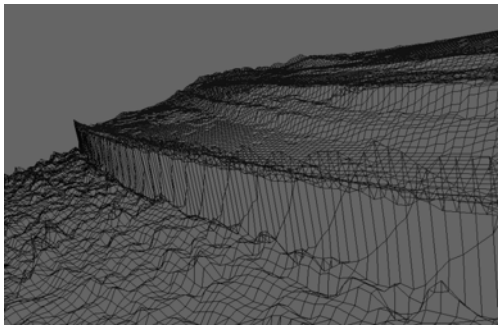


Figure 7
Illustration of an image surface before the deformation

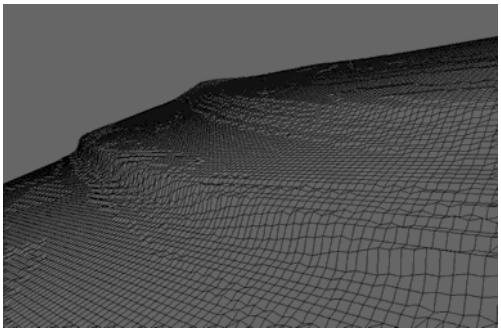


Figure 8
Illustration of an image surface after the deformation

Figure 9
Original image taken of a car

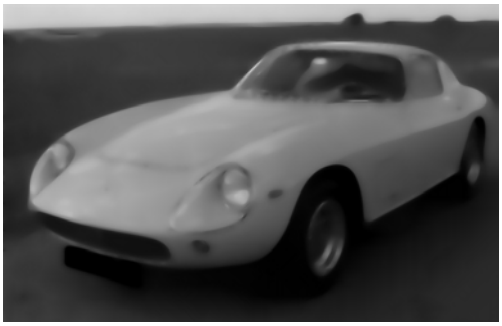


Figure 10
Smoothed image using surface deformation based on mean curvature

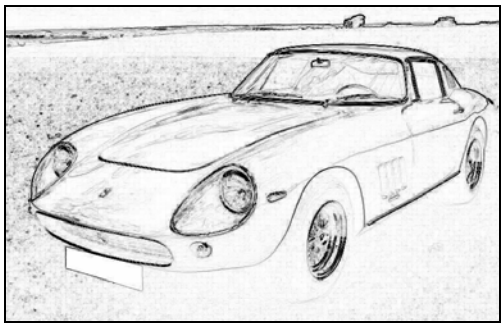


Figure 11
Edge map of the original image

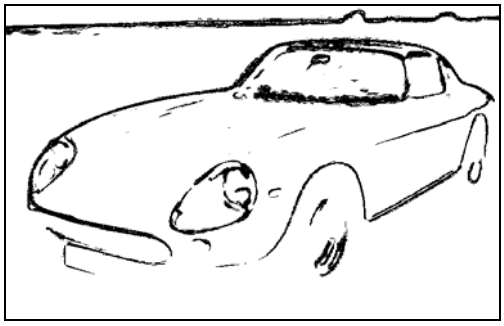


Figure 12
Edges after applying the proposed information enhancement method

5 3D Model Estimation from Multiple Images

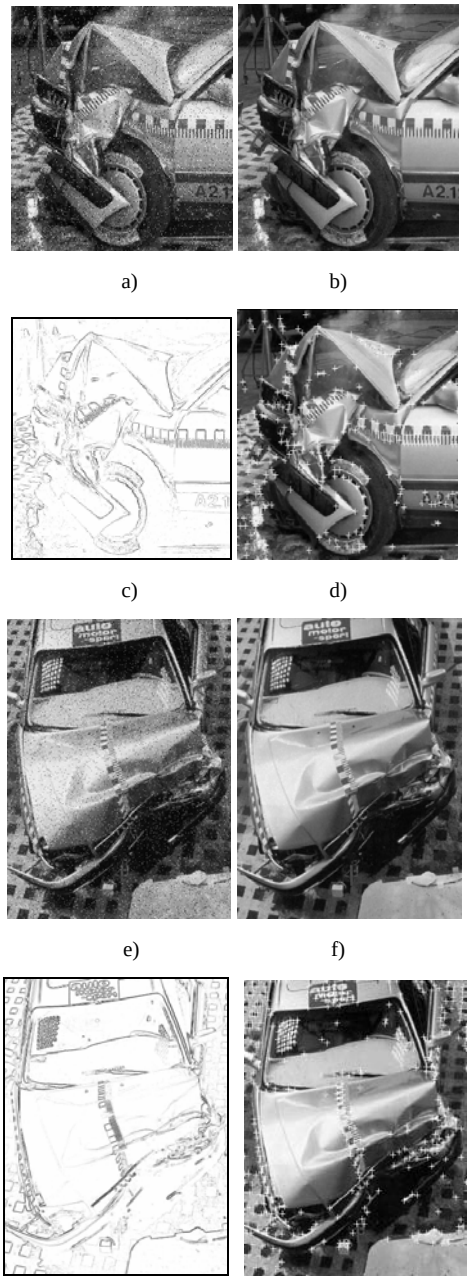
The basic concept of the 3D model estimation method described in the first part can be summarized as follows: As the first step, the pictures, used in the 3D-object reconstruction are preprocessed, which starts with noise elimination and edge detection by applying the fuzzy filters and fuzzy edge detection algorithm described in [24], [25]. This is usually followed by the primary edge extraction method (see Section V) [38].

For the modeling the determination of the primary edges and corners are very important because they carry the most characteristic information about the shape of the objects to be modeled. The applied corner detection method utilizes that a corner is indicated by two strong edges. It also applies fuzzy reasoning and the used local structure matrix composed of the partial derivatives of the gray level intensity of the pixels is extended by fuzzy decision making. The algorithm assigns also a new attribute, the fuzzy measure of being a corner, to the analyzed pixel. This property of the corners can advantageously be used at the searching for the corresponding corner points in stereo image pairs.

The next step is the determination of the 3D coordinates of the extracted edge points. First the corner point correspondences are determined which is followed by the determination of the edge correspondences in the different images. If the angle between the camera positions is relatively small then after the estimation of the projection matrices of the images (necessary for the calibration) the corresponding points can be calculated automatically with high reliability in each image. We search for the characteristic corner or edge points lying (in fuzzy sense) on the epipolar line and then the point correspondence matching is done by minimizing the fuzzy measure of the differences of the environment of the points with the help of a fuzzy supported searching algorithm [36]. The similarity of the above mentioned 'cornerness' is also considered. (The corresponding corner points keep their 'cornerness' property in the pictures near to each other with high reliability). Having the point correspondences we can calculate the 3D position of the image points (the camera calibration is solved by the determination of the Perspective Projection Matrix [32]) and in the knowledge of the 3D coordinates and the correspondences of the significant points the spatial model of the car body can easily be built.

The effectivity of the above 3D reconstruction method detailed in Section III of this paper is illustrated by a simple example.

Figure 13a shows the original photo of the crashed car corrupted by noise. In Figure 13b the fuzzy filtered image while in Figs. 13c and 13d the images after fuzzy based edge and corner detection can be followed. Figs. 13e-13h illustrate a different camera position of the car. The 3D model of the deformed part of the car-body is shown in Figure 14.



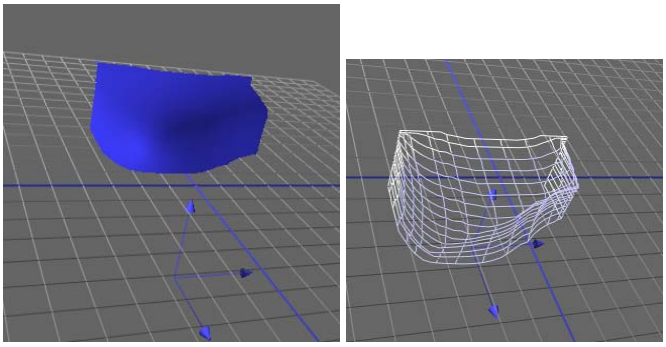
a) b)

c) d)

e) f)

g) h)

Figure 13
2 examples of the (a), (e): original photos, (b), (f): fuzzy filtered images, (c), (g): results after edge and (d), (h): corner detection of a crashed Audi 100



a) b)

Figure 14
3D model of the deformed part of the car body

6 Car Crash Analysis

In this Section a possible application of the introduced methods taken from vehicle system dynamics will be presented. The system aims the intelligent analysis of crashed cars and is able to determine the 3D model, the amount of the energy absorbed by the deformation and further important information, e.g. the energy equivalent speed and the direction of impact of the crash.

The block structure of the proposed new car crash analysis system can be followed in Figure 9. It contains four well defined sub-blocks. The first (image processing) is responsible for the pre-processing of the digital photos (noise elimination/filtering, edge detection, corner detection) and for the 3D modeling (including the point correspondence matching and the 3D model building). The second part of the system (comparison of models) calculates the volumetric change of the car body from the deformed and the original 3D models of the car. Parallel with it an expert system (Expert system) determines the direction of the impact. Based on the direction of impact and volumetric change a hierarchical fuzzy-neural network system (Fuzzy-Neural Network) determines the absorbed energy and the energy equivalent speed of the car. In the followings we will briefly outline the steps of the analysis not discussed previously.

After constructing the 3D model of the deformed car body (see Figure 8) we have to determine the volume of the detoriated car body which means that it is necessary to compare the deformed and the undamaged 3D car-bodies. This calculation is performed by the module named ‘Comparison of models’ (see Figure 15). The inputs of this module are the spatial models of the damaged and undamaged car-bodies. As result, we obtain the volumetric difference between the two models.

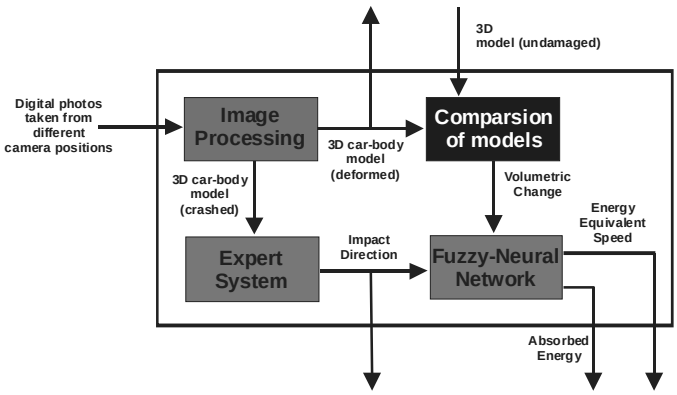


Figure 15
Block-structure of the intelligent car crash analysis system

The spatial model of the deformed car-body serves as input of another module, as well. This module applies an expert system and produces the direction of impact. For this we use the so called 'energy-centers' of the undamaged and deformed car-body parts and the direction is estimated from the direction of movement of the energy-center. (During the deformation the different 3D cells of the car-body absorb a certain amount of energy. The energy-center can be determined by weighting the cells by the corresponding energy values.)

From the volumetric difference and from the direction of impact an intelligent hierarchical fuzzy-neural network system evaluates the energy absorbed by the deformation and the equivalent energy equivalent speed (EES). For the training of this part of the system simulation and crash test data can be used. The training data include the volumetric change, the direction of the impact (input data) and the corresponding deformation energy (output data).

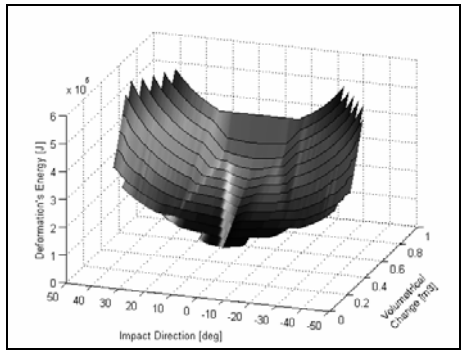


Figure 16
Relation among the direction of impact, volumetric change, and the deformation energy based on simulation data (Mercedes 290)

The relation among the direction of impact, volumetric change, and the deformation energy is illustrated by Figure 16. If, as usually is the case, this surface is symmetric (to the longitudinal axes of the vehicle), it is enough to deal with its half part. The mapping is approximated by a hierarchical fuzzy-NN system (subsystem 'Fuzzy-Neural Network' in Figure 15). The surface is divided into domains, which can 'easily' be modeled. Each domain is modeled separately by a small NN. Because of the uncertainties in the transitions among the domain, a fuzzy system is applied for the determination of the fired domain(s). The mapping in Figure 16 needs only to be divided into two domains according to the impact direction (see Fig. 17), thus in this very simple case the fuzzy rulebase 'above' the NN system contains only two rules (The input fuzzy sets are shown in Fig. 18):

IF the direction *IS* D1 *THEN* use NN1

IF the direction *IS* D2 *THEN* use NN2

Here we would like to remark two things:

1 In general the mapping is more complex and it can be advantageous to define more domains using both inputs to keep the complexity of the used NNs low.

2 The module responsible for the determination of the absorbed energy applies a pre-classification step according to a hierarchical decision-tree (Figure 19), because for choosing the correct set of neural networks we have to pre-determine the category and the type of the analyzed vehicle and the main character of the crash (frontal full impact, frontal offset impact, side impact, corner impact, rear impact). Cars are categorized into car types according to their weights. (In this paper as example a crashed Audi is shown. Although, the analysis is based on the NNs taught by the simulation data of a similar, but Mercedes car). Side impact means that neither the front nor the rear of the vehicle is touched.

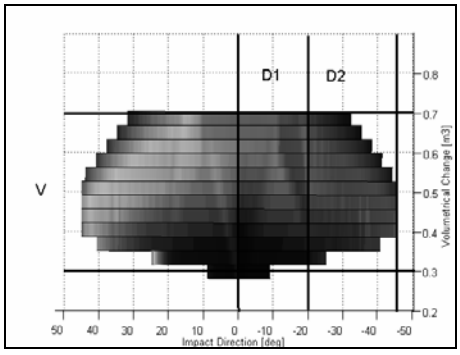


Figure 17
Segmentation of the surface in Figure 10

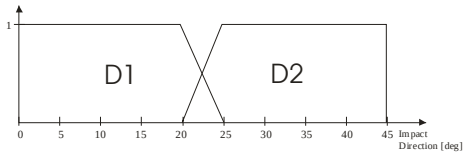


Figure 18
Membership functions defined on the universe of impact direction

For approximating domains D1 and D2 we applied simple feed-forward backpropagation NNs with one hidden layer and three hidden neurons. The NNs are used to determine the deformation's energy and EES. During the tuning (teaching period) of the system, the determined EES values were compared to known test results and the parameters of the expert system were modified to minimize the LMS error.

The operation of the introduced intelligent crash analysis system is illustrated on a crashed car. The parameters of the car are as follows:

Vehicle/Mass of the vehicle: Audi 100/1325 kg

Volumetric change (evaluated): 0.62 m^3

Absorbed deformation energy (evaluated): 171960 Joule

The resulted 3D model is shown in Figure 14. The results of the analysis are summarized in Table 1 (see also [39]). The error of the analysis depends on the resolution of the model (i.e. the distance between two layers in the 3D model, Figure 14b) and also on the accuracy of the crash test data.

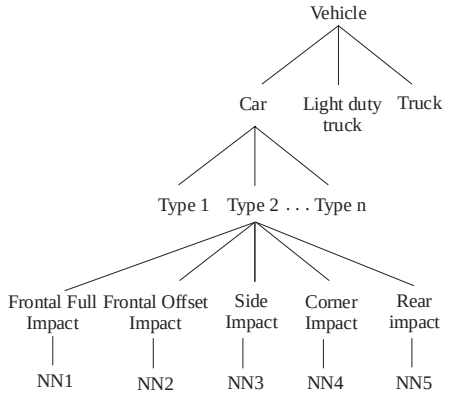


Figure 19
Hierarchical structure of the pre-classification in the EES determination

Table 1
The direction of impact and the energy equivalent speed of the crashed car

	Direction of impact [deg]	EES of the vehicle [km/h]
Real Data	0	55
Proposed method	2	58
2D method	2	59,5

Conclusions

In this paper intelligent methods are introduced which open a way for autonomous 3D model reconstruction. The 3D model reconstruction uses as input only digital images taken from different camera positions. The technique combines recent results of epipolar geometry, intelligent methods of image processing, and different fuzzy techniques. It applies a new edge information extraction procedure, as well, which is able to separate the edges carrying primary information and those representing only information of minor importance. The methods presented in the paper can advantageous be used in many 2D and 3D applications, in computer vision, in sketch based image retrieval methods, in vehicle system dynamics, etc.

As a possible new application taken of the field of vehicle system dynamics, an intelligent expert system is also presented which includes significant steps towards the autonomous analysis of car-crashes. It makes easy to determine the special shape of crashed cars (or other objects), the amount of the energy absorbed by the deformation, and further important information, like the energy equivalent speed (EES).

Acknowledgment

This work was sponsored by the Hungarian Fund for Scientific Research (OTKA T049519) and the Structural Fund for Supporting Innovation in New Knowledge and Technology Intensive Micro- and Spin-off Enterprises (GVOP-3.3.1-05/1.2005-05-0160/3.0)

References

[1] Taylor, C., P. Debevec, and J. Malik, “Reconstructing Polyhedral Models of Architectural Scenes from Photographs,” *Computer Vision - ECCV'96*, Lecture Notes in Computer Science, Vol. 1065/II, pp. 659-668, 1996

[2] Hartley, R., “Euclidean Reconstruction from Uncalibrated Views,” in J. L. Mundy, A. Zisserman, and D. Forsyth (eds.), *Applications of Invariance in Computer Vision*, Lecture Notes in Computer Science, Vol. 825, Springer-Verlag, pp. 237-256, 1994

[3] Hartley, R. and A. Zisserman, “*Multiple View Geometry in Computer Vision*,” Cambridge University Press, 2000

[4] Russo, F., "Fuzzy Filtering of Noisy Sensor Data," *In Proc. of the IEEE Instrumentation and Measurement Technology Conference*, Brussels, Belgium, 4-6 June 1996, pp. 1281-1285

[5] Russo, F., "Edge Detection in Noisy Images Using Fuzzy Reasoning," IEEE Transactions on Instrumentation and Measurement, Vol. 47, No. 5, Oct. 1998, pp. 1102-1105

[6] Russo, F., "Recent Advances in Fuzzy Techniques for Image Enhancement," *IEEE Transactions on Instrumentation and Measurement*, Vol. 47, No. 6, Dec. 1998, pp. 1428-1434

[7] Rogers D. F., *Procedural Elements for Computer Graphics,* McGraw Hill, New York, 1985

[8] Happer A., M. Araszewski, Practical Analysis Technique for Quantifying Sideswipe Collisions, 1999

[9] Melander, A., "Finite Element Simulation of Crash Testing of Laser Welded Joints," Research report, Swedish Institute for Metals, no: IM-2000-062, 2000

[10] Chen, H. F., C. B. Tanner, N. J. Durisek, and D. A. Guenther, "Pole Impact Speeds Derived from Bilinear Estimations of Maximum Crush for Body-On-Frame Constructed Vehicles," Paper no. 2004-01-1615, Society of Automotive Engineers, Warrendale, Pennsylvania, 2004

[11] C. Lu, Y. Cao, D. Mumford, "Surface Evolution under Curvature Flows", Submitted for the special issue on Partial Differential Equations (PDE's) in Image Processing," Computer Vision, and Computer Graphics, p. 19, 2002

[12] Gray, A. "The Gaussian and Mean Curvatures" and "Surfaces of Constant Gaussian Curvature," §16.5 and Ch. 21 in Modern Differential Geometry of Curves and Surfaces with Mathematica, 2nd ed. Boca Raton, FL: CRC Press, pp. 373-380 and 481-500, 1997

[13] Pollefeys, M., *Self-Calibration and Metric 3D Reconstruction from Uncalibrated Image Sequences*, PhD thesis, ESAT-PSI, K.U. Leuven, 1999

[14] Várkonyi-Kóczy, A. R., A. Rövid, "Fuzzy Logic Supported Corner Detection," *Journal of Intelligent and Fuzzy Systems*, to be published in 2007

[15] W. Förstner, "A Feature Based Correspondence Algorithm for Image Matching," *Int. Arch. Photogramm. Remote Sensing*, Vol. 26, pp. 150-166, 1986

[16] Rövid, A., A. R. Várkonyi-Kóczy, "Corner Detection in Digital Images Using Fuzzy Reasoning," *In Proc. of the 2nd IEEE International Conference on Computational Cybernetics*, August 30-Sept. 1, 2004, Vienna, Austria, pp. 95-99

[17] Várkonyi-Kóczy, A. R., A. Rövid, "Improved Fuzzy Based Corner Detection Method," *In Proc. of the IEEE Int. Workshop on Soft Computing Appl., SOFA'2005*, Szeged-Arad, Hungary-Romania, Aug. 27-30, 2005, pp. 237-242

[18] Várkonyi-Kóczy, A. R., A. Rövid, "Point Correspondence Matching for 3D Reconstruction Using Fuzzy Reasoning," *In Proc. of the 3rd IEEE Int. Conf. on Computational Cybernetics, ICCC 2005*, Mauritius, Apr. 13-16, 2005, pp. 87-92

[19] Rövid, A., A. R. Várkonyi-Kóczy, P. Várlaki, "3D Model Estimation from Multiple Images," *In Proc. of the IEEE International Conference on Fuzzy Systems, FUZZ-IEEE'2004*, July 25-29, 2004, Budapest, Hungary, Vol. 3, pp. 1661-1666

[20] Taylor, C., P. Debevec, J. Malik, "Reconstructing Polyhedral Models of Architectural Scenes from Photographs," *Computer Vision - ECCV'96*, Lecture Notes in Computer Science, Vol. 1065/II, pp. 659-668, 1996

[21] Hartley, R., "Euclidean Reconstruction from Uncalibrated Views," in J. L. Mundy, A. Zisserman, and D. Forsyth (eds.), *Applications of Invariance in Computer Vision*, Lecture Notes in Computer Science, Vol. 825, Springer-Verlag, pp. 237-256, 1994

[22] Hartley, R., A. Zisserman, "*Multiple View Geometry in Computer Vision*," Cambridge University Press, 2000

[23] Russo, F., "Fuzzy Filtering of Noisy Sensor Data," *In Proc. of the IEEE Instrumentation and Measurement Technology Conference*, Brussels, Belgium, 4-6 June 1996, pp. 1281-1285

[24] Russo, F., "Edge Detection in Noisy Images Using Fuzzy Reasoning," IEEE Transactions on Instrumentation and Measurement, Vol. 47, No. 5, Oct. 1998, pp. 1102-1105

[25] Russo, F., "Recent Advances in Fuzzy Techniques for Image Enhancement," *IEEE Transactions on Instrumentation and Measurement*, Vol. 47, No. 6, Dec. 1998, pp. 1428-1434

[26] Rogers D. F., *Procedural elements for Computer Graphics,* McGraw Hill, New York, 1985

[27] Happer A., M. Araszewski, Practical Analysis Technique for Quantifying Sideswipe Collisions, 1999

[28] Melander, A., "Finite Element Simulation of Crash Testing of Laser Welded Joints," Research report, Swedish Institute for Metals, no: IM-2000-062, 2000

[29] Chen, H. F., C. B. Tanner, N. J. Durisek, and D. A. Guenther, "Pole Impact Speeds Derived from Bilinear Estimations of Maximum Crush for Body-

On-Frame Constructed Vehicles," Paper no. 2004-01-1615, Society of Automotive Engineers, Warrendale, Pennsylvania, 2004

[30] C. Lu, Y. Cao, D. Mumford, "Surface Evolution under Curvature Flows", Submitted for the special issue on Partial Differential Equations (PDE's) in Image Processing," Computer Vision, and Computer Graphics, p. 19, 2002

[31] Gray, A. "The Gaussian and Mean Curvatures" and "Surfaces of Constant Gaussian Curvature," §16.5 and Ch. 21 in Modern Differential Geometry of Curves and Surfaces with Mathematica, 2nd ed. Boca Raton, FL: CRC Press, pp. 373-380 and 481-500, 1997

[32] Pollefeys, M., *Self-Calibration and Metric 3D Reconstruction from Uncalibrated Image Sequences*, PhD thesis, ESAT-PSI, K.U. Leuven, 1999

[33] W. Förstner, "A Feature Based Correspondence Algorithm for Image Matching," *Int. Arch. Photogramm. Remote Sensing*, Vol. 26, pp. 150-166, 1986

[34] Rövid, A., A. R. Várkonyi-Kóczy, "Corner Detection in Digital Images Using Fuzzy Reasoning," *In Proc. of the 2nd IEEE International Conference on Computational Cybernetics*, August 30-Sept. 1, 2004, Vienna, Austria, pp. 95-99

[35] Várkonyi-Kóczy, A. R., A. Rövid, "Improved Fuzzy Based Corner Detection Method," *In Proc. of the IEEE Int. Workshop on Soft Computing Appl., SOFA'2005*, Szeged-Arad, Hungary-Romania, Aug. 27-30, 2005, pp. 237-242

[36] Várkonyi-Kóczy, A. R., A. Rövid, "Point Correspondence Matching for 3D Reconstruction Using Fuzzy Reasoning," *In Proc. of the 3rd IEEE Int. Conf. on Computational Cybernetics, ICCC 2005*, Mauritius, Apr. 13-16, 2005, pp. 87-92

[37] Rövid, A., A. R. Várkonyi-Kóczy, P. Várlaki, "3D Model Estimation from Multiple Images," In Proc. of the IEEE International Conference on Fuzzy Systems, FUZZ-IEEE'2004, July 25-29, 2004, Budapest, Hungary, Vol. 3, pp. 1661-1666

[38] Rövid, A., T. Hashimoto, A. R. Várkonyi-Kóczy, Y. Shimodaira, "Information Enhancement Method for Image Retrieval and Object Recognition," In Proc. of the 3rd Int. Symposium on Computational Intelligence and Intelligent Informatics, ISCIII 2007, Agadir, Morocco, March 28-30, pp. 25-29

[39] Várkonyi-Kóczy, A. R., A. Rövid, M. G. Ruano, "Soft Computing Based Car Body Deformation and EES Determination for Car Crash Analysis Systems," IEEE Trans. on Instrumentation and Measurement, Vol. 55, No. 4, August 2006