



“SERVICLE” – Smart System to Detect Dents and Scratches in Vehicle Body

Dr. Shantakumar B Patil^[1] Dr. Premjoti Patil^[2] Mrs. Gouri Patil^[3] Mr. Mahesh B Patil^[4]

^{[1][2]} Professor

Nagarjuna College of Engineering & Technology, Bengaluru, Karnataka.

^[3] Assistant Professor

Muffakham Jah College of Engineering and Technology (MJCET).

^[4] Assistant Professor

SDM Institute of Technology -[SDMIT] Ujire.

shantakumar.p@gmail.com[1]

1. ABSTRACT:

Automatic minor damage recognition for vehicles (e.g. recognition of dents and scratches) allows car rental and car sharing companies to assign damage to a guilty customer. This can be done by comparing the point of time when the damage occurs with the booking time of customers. If necessary, the companies inform the customer about an event during his booking time. In particular, the damage detection system creates the necessary transparency between the customer and claims management in the car sharing business. The philosophy of car sharing and the growing number of users sets high requirements to the check-out procedure during usage. These worsen the relationship between customer and car sharing company, since there is a possibility that the damage is assigned to a wrong person.

Keywords: Servicle, Detection, Dents

2. INTRODUCTION:

The presented approach deals with the algorithm development for minor damage identification in vehicle bodies using adaptive sensor data processing. It ensures the correct and error-free classification of the detected damage events, and increases the credibility of the damage messages.

A new automatic dent detection system which can automatically obtain the surface information of the car bodies, extract features, reduce noise effects and distinguish dents from features, noise and detect the dents, scratches and broken glass. The system will also

generate position information that is mark the particular dent, scratch, broken glass area.

First stage is car reception where the system will take pictures to the car from its different sides by using

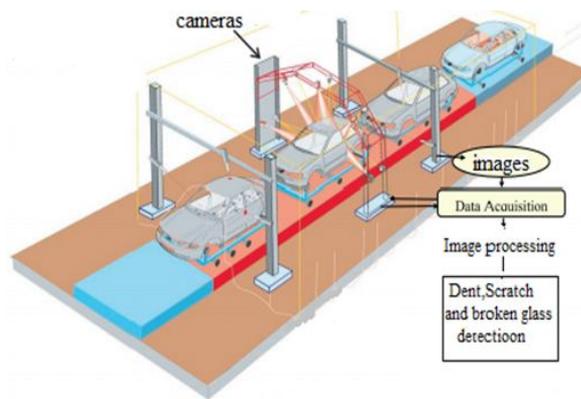


Figure 1: Proposed system

3. SYSTEM ARCHITECTURE

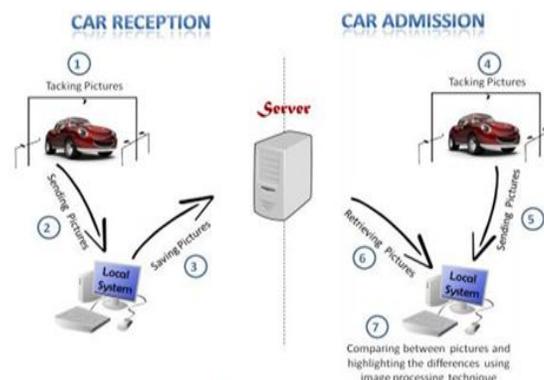


Figure 2: System Architecture Diagram of the proposed model

professional cameras and assemble all these images to form 3-D images by using Blender software. Second stage is car admission where the images will be uploaded into the system and the original images of the car is automatically displayed on the screen in order to compare it with the pictures that are captured during the arrival for inspection using image processing technique. If there are any differences between the pictures, the system highlights them directly. The database will be updated with the new pictures for the car and finally a report with the inspection details will be reported to the customers.

4. IMAGE PROCESSING

Digital image processing refers processing of the image in digital form. Modern cameras may directly take the image in digital form but generally images are originated in optical form. They are captured by video cameras and digitalized. The digitalization process includes sampling, quantization. Then these images are processed by the five fundamental processes, at least any one of them, not necessarily all of them.

5. IMAGE ACQUISITION

Image is read into MATLAB using the command `imread('image_name.jpg')`

The `imread` function supports four general syntaxes, described below. The `imread` function also supports several other format-specific syntaxes.

`A = imread(filename,fmt)` reads a grayscale or color image from the file specified by the string `filename`, where the string `fmt` specifies the format of the file. If the file is not in the current directory or in a directory in the MATLAB path, specify the full pathname of the location on your system. For a list of all the possible values for `fmt`, see Supported Formats. If `imread` cannot find a file named `filename`, it looks for a file named `filename.fmt`.

`imread` returns the image data in the array `A`. If the file contains a grayscale image, `A` is a two-dimensional (M-by-N) array. If the file contains a color image, `A` is a three-dimensional (M-by-N-by-3) array. The class of the returned array depends on the data type used by the file format. See Class Support for more information.

For most file formats, the color image data returned uses the RGB color space. For TIFF files, however, `imread` can return color data that uses the RGB, CIELAB, ICCLAB, or CMYK color spaces. If the color image uses the CMYK color space, `A` is an M-by-N-by-4 array. See the TIFF-Specific Syntax for more information.

`[X,map] = imread(filename,fmt)` reads the indexed image in `filename` into `X` and its associated colormap into `map`. The colormap values are rescaled to the range [0,1].

`[...] = imread(filename)` attempts to infer the format of the file from its content.

`[...] = imread(URL,...)` reads the image from an Internet URL. The URL must include the protocol type (e.g., `http://`).

6. RGB2GRAY CONVERSION

Typical RGB input devices are color TV and video cameras, image scanners, and digital cameras. Typical RGB output devices are TV sets of various technologies (CRT, LCD, plasma, etc.), computer and mobile phone displays, video projectors, multicolor LED displays, and large screens such as JumboTron. Color printers, on the other hand, are not RGB devices, but subtractive color devices (typically CMYK color model). The need for converting to Grayscale is to reduce the processing time for the Algorithms, RGB Images are not required to Processing.

$$\text{intensity} = 0.2989 \cdot \text{red} + 0.5870 \cdot \text{green} + 0.1140 \cdot \text{blue}$$

7. GRAYSCALE

Technical uses (e.g. in medical imaging or remote sensing applications) often require more levels, to make full use of the sensor accuracy (typically 10 or 12 bits per sample) and to guard against roundoff errors in computations. Sixteen bits per sample (65,536 levels) is a convenient choice for such uses, as computers manage 16-bit words efficiently. The TIFF and the PNG (among other) image file formats support 16-bit grayscale natively, although browsers and many imaging programs tend to ignore the low order 8 bits of each pixel. No matter what pixel depth is used, the binary representations assume that 0 is black and the maximum value (255 at 8 bpp, 65,535 at 16 bpp, etc.) is white, if not otherwise noted.



Figure 3: Gray scale Image

8. IMAGE RESIZING

In computer graphics, image scaling is the process of resizing a digital image. Scaling is a non-trivial process that involves a trade-off between efficiency, smoothness and sharpness. With bitmap graphics, as the size of an image is reduced or enlarged, the pixels that form the image become increasingly visible, making the image appear "soft" if pixels are

averaged, or jagged if not. With vector graphics the trade-off may be in processing power for re-rendering the image, which may be noticeable as slow re-rendering with still graphics, or slower frame rate and frame skipping in computer animation.

`B = imresize(A, scale)` returns image B that is scale times the size of A. The input image A can be a grayscale, RGB, or binary image. If scale is from 0 through 1.0, B is smaller than A. If scale is greater than 1.0, B is larger than A. By default, `imresize` uses bicubic interpolation.



Figure 4: Image resize

9. GRAYSCALE TO BINARY CONVERSION

`BW = im2bw(I, level)` converts the grayscale image I to a binary image. The output image BW replaces all pixels in the input image with luminance greater than level with the value 1 (white) and replaces all other pixels with the value 0 (black). Specify level in the range [0,1]. This range is relative to the signal levels possible for the image's class. Therefore, a level value of 0.5 is midway between black and white, regardless of class. To compute the level argument, you can use the function `graythresh`. If you do not specify level, `im2bw` uses the value 0.5. `BW = im2bw(X, map, level)` converts the indexed image X with colormap map to a binary image. `BW = im2bw(RGB, level)` converts the truecolor image RGB to a binary image. If the input image is not a grayscale image, `im2bw` converts the input image to grayscale, and then converts this grayscale image to binary by thresholding.

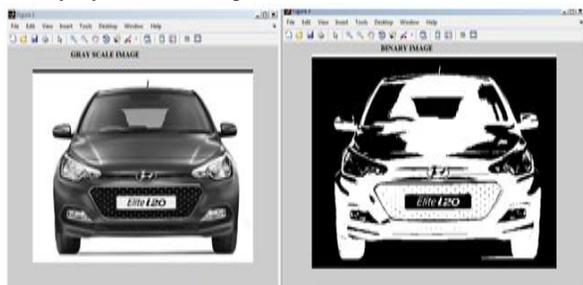


Figure 5: Grey scale to binary conversion

10. BW-BOUNDARIES / BINARISATION

`B = bw-boundaries(BW)` traces the exterior boundaries of objects, as well as boundaries of holes inside these objects, in the binary image BW. `Bw-boundaries` also descends into the outermost objects (parents) and traces

their children (objects completely enclosed by the parents). BW must be a binary image where nonzero pixels belong to an object and 0 pixels constitute the background. The following figure illustrates these components.

Recursive Gaussian filter algorithm The classical method of smoothing an image by convolution with a Gaussian kernel has the drawback that it is slow when the standard deviation σ of the Gaussian is large. This is due to the larger size of the kernel, which results in a higher number of computations per pixel. The `itk::RecursiveGaussianImageFilter` implements an approximation of convolution with the Gaussian and its derivatives by using IIR2 filters. In practice this filter requires a constant number of operations for approximating the convolution, regardless of the σ value [21, 22].

This filter applies the approximation of the convolution along a single dimension. It is therefore necessary to concatenate several of these filters to produce smoothing in all directions. In this example, we create a pair of filters since we are processing a 2D image. Since each one of the newly created filters has the potential to perform filtering along any dimension, we have to restrict each one to a particular direction. This is done with the `SetDirection()` method.

11. ADVANCED 3D IMAGE PROCESSING TECHNIQUE

There are two typical ways of normalizing Gaussians depending on their application. For scale-space analysis it is desirable to use a normalization that will preserve the maximum value of the input. This normalization is represented by the following equation.

$$1 / \sigma \sqrt{2\pi}$$

In applications that use the Gaussian as a solution of the diffusion equation it is desirable to use a normalization that preserve the integral of the signal. This last approach can be seen as a conservation of mass principle. This is represented by the following equation.

$$1 / \sigma^2 \sqrt{2\pi}$$

The input image can be obtained from the output of another filter. Here, an image reader is used as the source. The image is passed to the x filter and then to the y filter. The reason for keeping these two filters separate is that it is usual in scale-space applications to compute not only the smoothing but also combinations of derivatives at different orders and smoothing. Some factorization is possible when separate filters are used to generate the intermediate results. Here this capability is less interesting, though,

since we only want to smooth the image in all directions.

It is now time to select the σ of the Gaussian used to smooth the data. Note that σ must be passed to both filters and that sigma is considered to be in millimeters. That is, at the moment of applying the smoothing process, the filter will take into account the spacing values defined in the image.

12.RESULTS

`B = bw-boundaries(BW, conn)` specifies the connectivity to use when tracing parent and child boundaries. `conn` can have either of the following scalar values.

`Bw-boundaries` returns `B`, a P-by-1 cell array, where P is the number of objects and holes. Each cell in the cell array contains a Q-by-2 matrix. Each row in the matrix contains the row and column coordinates of a boundary pixel. Q is the number of boundary pixels for the corresponding region.

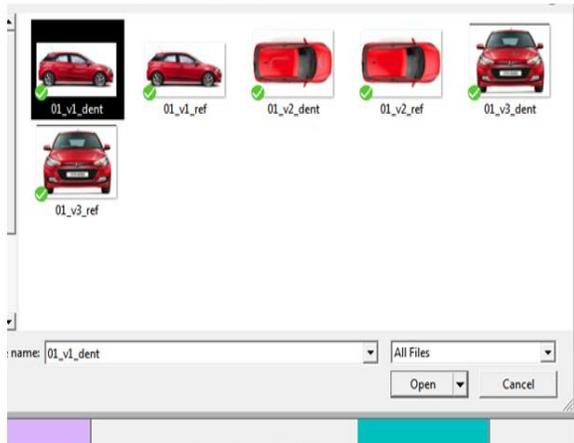
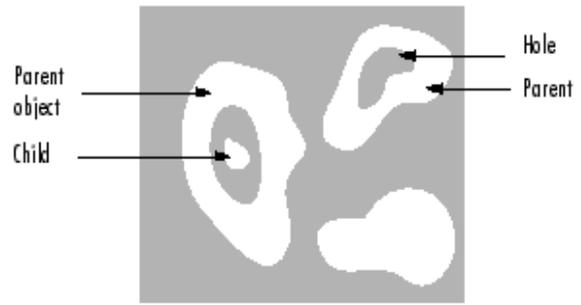


Figure 7: The detected dents and the scratches of all the views in a single image



11. BLENDER:

With the help of this software 3D image of the car is created. Same 3D figure is stored in the database and copy is send to the customer mobile.

12. CONCLUSION:

The main purpose of our project is to increase the detection performance and it should have less computational cost, or comparable cost when compared with the existing approaches. The proposed hierarchical INCs detection approach is fast, adaptive, and fully automatic. The presented CADs system should yield comparable detection accuracy and more computational efficiency than existing systems, which should be use for clinical utility. This low-level VQ illustrates adequate detection power for non GGO nodules, and is computationally more efficient than the state of-the-art approaches. It is possible to detect the damages on the entire vehicle body with the help of specific algorithms. The server periodically receives the status of the system and stores the received information of the reported damages.

The presented approach deals with the algorithm development for minor damage identification in vehicle bodies using adaptive sensor data processing. It ensures the correct and error-free classification of the detected damage events, and increases the credibility of the damage messages. This can be enhanced by capturing the 3D images and detecting the dents and the scratches on the car body. This can be enhanced by sending the detailed report to the customer’s mobile.

REFERENCES:

- [1] “Dent Detection in Car Bodies” by T. Lilienblum, P. Albrecht, R. Calow, B. Michaelis
- [2] “Investigations and Comparison of Noise Signals to Useful Signals for the Detection of Dents in Vehicle Bodies by Sound Emission Analysis” by Hauke Baumgartel, Andre Kneifel, Sergei Gontcharov, Karl-Ludwig Krieger.

- [3] “Visualization and Detection of Small Defects on Car-Bodies” by S. Karbacher, J. Babst, G. Häusler, X. Laboureux
- [4] “Algorithm Development for Minor Damage Identification in Vehicle Bodies Using Adaptive Sensor Data Processing” by Sergei Gontcharov, Hauke Baumgärtel, Karl-Ludwig Krieger.
- [5] Baumgärtel H, Skwarek V, Kneifel A, Gontcharov S, Krieger K-L. Intelligent monitoring of vehicle body damages for CarSharing applications. *Electronics in Vehicles*, Baden-Baden, 2013, Bd. 16. Hauke Baumgärtel et al. / *Procedia Technology*
- [6] Kneifel A, Harmstorf J, Baumgärtel H, Gontcharov S, Krieger K-L. Localizing Minor Damages in Cars by means of Structure-Borne Sound Analysis. 14th Stuttgart International Symposium, Stuttgart, 2014.
- [7] Gontcharov S, Krieger K-L, Kneifel A, Baumgärtel H, Hellbernd B, Marz V. Verteilte Intelligenz in einem Sensornetzwerk zur Körperschalldetektion in Fahrzeugen. *Embedded Systems Symposium*, München, 2013.
- [8] Karbacher S, Babst J, Häusler G, Laboureux X. Visualization and Detection of Small Defects on Car-Bodies. University of Erlangen, 1999.
- [9] Lilienblum T, Albrecht P, Calow R, Michaelis B. Dent Detection in Car Bodies. Otto von Guericke University, Magdeburg, 2000.
- [10] Yogeswaran A, Payeur P. 3D Surface Analysis for Automated Detection of Deformations on Automotive Body Panels. *New Advances in Vehicular Technology and Automotive Engineering*, Chapter 12, InTech, ISBN 978-953-51-0698-2, 2012.
- [11] Müller M. Detektion und Klassifikation von Krafteinwirkungen auf die Fahrzeugaußenhaut auf Grundlage piezoelektrischer Signale. Phd thesis. University of Rostock; 2013.
- [12] Baumgärtel H, Kneifel A, Gontcharov S, Skwarek V, Krieger K-L. Drei-kanaliger piezoelektrischer Sensor zur Detektion von Bagatellschäden und deren Ursprungsort an Fahrzeugkarosserieteilen. *Sensoren im Automobil*, München, 2014
- [13] Kammeyer KD, Kroschel K. *Digitale Signalverarbeitung*. 7th Edition. Wiesbaden: Vieweg Teuber Verlag; 2009.
- [14] The MathWorks Inc. Envelope Detection. <http://www.mathworks.de/de/help/dsp/examples/envelope-detection.html>, Application note, accessed Jan. 2014.
- [15] H. Baumgärtel, V. Skwarek, A. Kneifel, S. Gontcharov, K.-L. Krieger Intelligent monitoring of vehicle body damages for CarSharing applications *Electronics in Vehicles*, Baden-Baden (2013) Bd. 16
- [16] A. Kneifel, J. Harmstorf, H. Baumgärtel, S. Gontcharov, K.-L. Krieger Localizing Minor Damages in Cars by means of Structure-Borne Sound Analysis 14th Stuttgart International Symposium. Stuttgart (2014)
- [17] S. Gontcharov, K.-L. Krieger, A. Kneifel, H. Baumgärtel, B. Hellbernd, V. Marz Verteilte Intelligenz in einem Sensornetzwerk zur Körperschalldetektion in Fahrzeugen *Embedded Systems Symposium*, München (2013)
- [18] S. Karbacher, J. Babst, G. Häusler, X. Laboureux Visualization and Detection of Small Defects on Car-Bodies University of Erlangen (1999)
- [19] T. Lilienblum, P. Albrecht, R. Calow, B. Michaelis Dent Detection in Car Bodies Otto von Guericke University, Magdeburg (2000)
- [20] Yogeswaran A, Payeur P. 3D Surface Analysis for Automated Detection of Deformations on Automotive Body Panels. *New Advances in Vehicular Technology and Automotive Engineering*, Chapter 12, InTech, ISBN 978-953-51-0698-2, 2012.
- [21] Müller M. Detektion und Klassifikation von Krafteinwirkungen auf die Fahrzeugaußenhaut auf Grundlage piezoelektrischer Signale. Phd thesis. University of Rostock; 2013.
- [22] H. Baumgärtel, A. Kneifel, S. Gontcharov, V. Skwarek, K.-L. Krieger Drei-kanaliger piezoelektrischer Sensor zur Detektion von Bagatellschäden und deren Ursprungsort an Fahrzeugkarosserieteilen *Sensoren im Automobil*, München (2014)
- [23] K.D. Kammeyer, K. Kroschel *Digitale Signalverarbeitung* (7th Edition), Vieweg Teuber Verlag, Wiesbaden (2009)
- [24] The MathWorks Inc. Envelope Detection. <http://www.mathworks.de/de/help/dsp/examples/envelope-detection.html>, Application note, accessed Jan. 2014.
- [25] A. Aamodt and E. Plaza. Casebased reasoning: Foundational issues, methodological variations, and system approaches. *Artificial Intelligence Communications*, 7(1):21–40, 1994.
- [26] P. J. Besl and R. C. Jain. Invariant surface characteristics for 3d object recognition in range images. *Computer Vision, Graphics, and Image Processing*, 33:33–80, 1986.
- [27] B. Dörband, R. Köthe, and A. Zilker. Streifenprojektion prüft in Echtzeit mit 5 µm Auflösung. *Werkstatt und Betrieb*, 128(3), 1995.
- [28] G. Häusler and K. Engelhardt. Vergleichende Studie zur automatisierten Lackfehlerbestimmung. *Expertise*, University of Erlangen-Nürnberg, August 1987.
- [29] G. Häusler and S. Karbacher. Reconstruction of smoothed polyhedral surfaces from multiple range images. In B. Girod, H. Niemann, and H.-P. Seidel, editors, *3D Image Analysis and Synthesis '97*, pages 191–198, Sankt Augustin, Germany, 1997. Infix Verlag.
- [30] S. Karbacher. Rekonstruktion und Modellierung von Flächen aus Tiefenbildern. Phd thesis, University of Erlangen-Nürnberg, Shaker Verlag, Aachen, 1997.
- [31] M. Kujawinska. Spatial phase measurement methods. In David W. Robinson and Graeme T. Reid, editors, *Interferogram Analysis: Digital Fringe Pattern Measurement Techniques*, pages 141–193. Institute of Physics Publishing, Bristol, 1993.

- [32] Sira Ltd. Car paint inspection system. Outline description, Chislehurst, U.K., April 1987.
- [33] H. Marguerre. Ein neues inkohärentes Schlierenverfahren mit Retroreflektor. *Optik*, 71(3):105–112, 1985.
- [34] R. L. Reynolds, F. Karpala, D. A. Clarke, and O. L. Hageniers. Theory and applications of a surface inspection technique using double-pass retroreflection. *Optical Engineering*, 32(9):2122–2129, 1993.