

Document downloaded from:

<http://hdl.handle.net/10251/101844>

This paper must be cited as:



The final publication is available at

<https://doi.org/10.1016/j.jmsy.2017.07.006>

Copyright Elsevier

Additional Information

Detecting Dings and Dents on Specular Car Body Surfaces based on Optical Flow

Laura Arnal, J. Ernesto Solanes¹, Jaime Molina and Josep Tornero

Universitat Politècnica de València, Institute of Design and Manufacturing, Camí de Vera S/N, 46022, València, Spain (Email address: {laurarbe,esolanes,jaimopar,jtornero}@idf.upv.es)

Abstract

This paper introduces a new approach to detect defects cataloged as dings and dents on car body surfaces, which is currently one of the most important issues facing quality control in the automotive industry. Using well-known optical flow algorithms and the deflectometry principle, the method proposed in this work is able to detect all kind of anomalies on specular surfaces. Hence, our method consists of two main steps: first, in the *pre-processing step*, light patterns projected on the body surface sweep uniformly the area of inspection, whilst a new image fusion law, based on optical flow, is used to obtain a resulting fused image holding the information of all variations suffered by the projected patterns during the sweeping process, indicating the presence of anomalies; second, a new *post-processing step* is proposed that avoids the need of using pre-computed reference backgrounds in order to differentiate defects from other body features such as style-lines. To that end, the image background of the resulting fused image is estimated in the first place through a method based on blurring the image according to the direction of each pixel. Afterwards, the estimated image background is used in a new subtraction law through which defects are well differentiated from other surface deformations, allowing the detection of defects in the entire illuminated area. In addition, since our approach, together with the system used, computes defects in less than 15 seconds, it satisfies the assembly plants time requirements. Experimental results presented in this paper are obtained from the industrial automatic quality control system *QEyeTunnel* employed in the production line at the Mercedes-Benz factory in

¹Corresponding author.

Vitoria, Spain. A complete analysis of the algorithm performance will be shown here, together with several tests proving the robustness and reliability of our proposal.

Keywords: Deflectometry, Optical Flow, Image fusion, Specular surfaces, Painted surfaces, Car Body inspection.

1. Introduction

Quality control in the automotive industry is one of the most important processes in manufacturing an automobile. The majority of the component parts that go into the automobile are produced at other sites. Outside parts vendors subject their component parts to rigorous testing and inspection audits to accomplish the quality standards of automobile manufacturers. Hence, the assembly plants can anticipate that the products arriving at their receiving docks are free from defects. Once the component parts of the car begin to be assembled at the automotive factory, quality audit stations in the assembly plants keep track of vital information concerning the integrity of various functional components of the vehicle.

One of the strictest of these quality auditions is the one located just after the painting process station. The reason is that the result of the painting process must meet the quality standards demanded by potential customers of the automobile company. For most of the brands, quality auditions after the painting process are carried out using manual detection and subjective evaluation by experts, known as *check-men* (shown in Fig. 1). In Fig. 2, some examples of defects that these experts face every day are shown. All of them are cataloged as critical, which means that *check-men* must ensure that none of them remain unrepaired. Notice the difficulty of determining the existence of defects in the examples shown. In fact, we need to rely on the deformation that a pattern of light projected on the surface suffers with the presence of an anomaly. This is known as *deflectometry principle* in the fields of optics and computer vision [1, 2, 3].

To accomplish the inspection task, quality audit stations after the painting process are provided by proper illumination that allows *check-men* to make use of the deflectometry principle (i.e., side, top and floor light bars put through along the station in



Figure 1: Check-men working on a production line, inspecting car body parts following the painting stage (image courtesy of Mercedes Benz España S.A.U., Vitoria Plant).

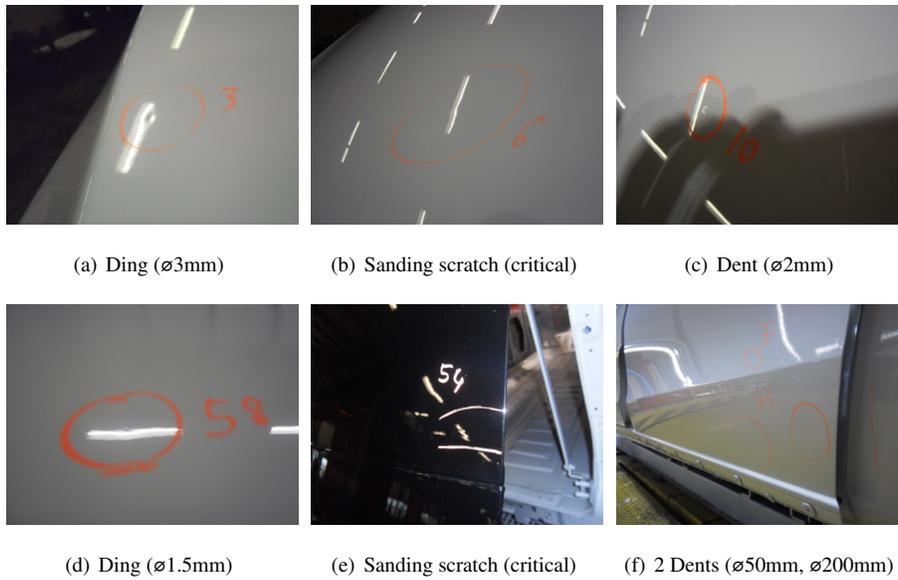


Figure 2: Defects in painted car body surfaces detected by Mercedes-Benz quality control experts (courtesy of Mercedes Benz España S.A.U., Vitoria Plant).

Fig. 1). Using this methodology, most of the current automobile companies can guarantee the following quality standards: detection and repairment of small defects less than $\varnothing 1\text{mm}$ and of up to $\varnothing 0.2\text{mm}$ (e.g., dirt, orange peel or pinholes); detection and repairment of all defects cataloged as dings and dents (e.g., scratches, drop drains and all kind of surface deformations greater than $\varnothing 1\text{mm}$).

There are some issues regarding to the current manual inspection process: first, the difference in evaluation criteria applied by *check-men* across the board, second, the difference in the evaluation criteria applied on an individual basis by *check-men* on any one day, and third, the restriction in the time check-men have to carry out the entire inspection (i.e., 30 seconds/body in case of passenger cars). These issues make it impossible to guarantee the same minimum quality standard for all vehicles given that the process is affected by human errors and, as a consequence, product quality managers in the automotive industry are still looking for a systematic solution.

The present work deals with the detection of defects cataloged as dings and dents. Let us turn our attention to existing research. For instance, in [4, 5] the authors present a technique to extract unwanted deformations on automotive body parts from a high resolution 3D mesh and to gather them based on their proximity and similarity using a single octree structure. The consistency of the approach using lower levels of resolution is not specified nor the computational cost of the method. Other different approach is the one presented in [6], where the detection is performed on sheet metal parts using deflectometry techniques in the infrared range. The method consists of projecting a known light pattern over the sheet surface, which projection is then captured by a thermo-camera. Afterwards, a Fourier-transform method is used to analyze the image and detect the defects. As the authors manifest, this approach is appropriate for the detection of defects on large, smooth specular reflecting objects. However, this paper does not detail the computational cost of the algorithm nor any prove of the performance using real car body parts. In addition, there is a trend in designing car models to highlight all the style-lines, making them sharper. This fact greatly reduces the detection area of these approaches.

It is possible to find approaches addressing the problem of defect detection on specular surfaces used in other fields. An example of this is found in the ceramic industry,

where the automatic detection of defects on painted tiles is being studied for the last decade. For instance, [7] proposed a low-cost system based on smart cameras to detect seeds defects. The approach is based on the fact that seeds defects can be modeled with a Gaussian model. This is hardly applicable to the automotive industry since defects of the same typology can vary in shape and size as it is shown in Fig. 2. Other interesting approach is the one presented in [8], whose authors made use of morphology techniques, such as well known edge detectors Sobel and Canny [9], to detect scratches on painted tiles. This work, however, does not detail how the noise affects to the performance of their method. The effect of noise is one aspect that must be considered when facing the problem of defect detection on car bodies. The amount of noise can differ considerably from one assembly plant to another, depending on several factors such as environmental light contamination, material properties of the sheet, paint used, the painting process itself, among others [10]. In addition to the commented issues, and contrary to what happens in car bodies surfaces, ceramic tiles are flat surfaces, which simplifies considerably the problem of detecting anomalies.

Other example is found in the plastic industry. Thus, [11] presented an approach based on local binary pattern operator together with the bidirectional reflectance distribution function in order to detect various surface defects and estimate their visibility. As it is pointed by authors, the presented algorithm needs to acquire data from different illumination angles. The comparison of the resulting intensity image with reference patterns allows the detection of defects. The approach is validated only with two flat surfaces without indicating which is the effect in terms of false positives or bad detections produced when there are position errors. Note that, in all systems based on the deflectometry principle, the relative pose between camera, surface and light can be different from one inspection to another. These positioning errors introduce intensity variations at pixel level between the reference patterns and each inspection. These differences are magnified even more with concave and convex surfaces such as those of car bodies, which can cause either false positives or loss of the detection quality.

Despite the commented research approaches and laboratory setups, automatic industrial inspection systems have been currently developed. For instance, [12] and [13] sell two similar systems, which use industrial robots, cameras and deflectometry tech-

niques to detect, among other kind of defects, some of the cataloged as dings and dents. The goal is to cover as much area as possible approaching the vision and illumination pattern system to the body surface, but it is not clear and well explained the details of the detection algorithms. In this sense, the non-inspected area can be reduced and the system could cover much more area rather other similar systems such as [10, 14]. The main problem of [12] is the cycle time restriction of the assembly plants. On the contrary, the systems developed in [10, 14] fulfill the requirements of the production lines in terms of cycle time and inspection quality for small defects. However, dings and dents are out of their detection capabilities.

In addition, some patents such as [15, 16, 17], can be found, proposing systems for the inspection and detection of defects on specular surfaces, none of which clarify or indicate how to overcome the problems of detecting dings and dents on painted car bodies.

Therefore, the aim of this work is to present a feasible industrial method for detecting defects cataloged as dings and dents on painted car bodies regardless their shape and structure (whether the paint is broken or the anomaly is big and smooth). Contrary to current approaches, which rely on intensity maps [10, 14, 12, 6], light pattern comparisons in the frequency domain [18], or 3D reconstructions [5], the proposed method yields in the sudden changes of velocity that a ray sweeping the surface suffers with the presence of an anomaly. This is carried out by using well-known image flow methods [19] and the deflectometry principle.

Hence, our method consists of two main steps: first, in the *pre-processing step*, light patterns projected on the body surface sweep uniformly the area of inspection, whilst a new image fusion law is used to obtain a resulting fused image holding the information of all velocities variations suffered by the projected patterns during the sweeping process, indicating the presence of anomalies; second, a new *post-processing step* is proposed which avoids the need of using pre-computed reference backgrounds to be able to differentiate defects from other body features. To that end, the background of the resulting fused image is estimated, in the first place, through a method based on blurring the image according to the direction of each pixel. Afterwards, the estimated background is used in a new subtraction law through which defects are well differenti-

ated from other surface deformations such as style lines, so the detection is performed in the entire illuminated area. Note that, contrary to most of current approaches, our method eliminates the need of using pre-computed backgrounds, avoiding miss matching problems that can produce false detections, and thus, increases the reliability of the detection. In addition, since our approach, together with the system used, computes defects in less than 15 seconds, it satisfies assembly plants time production requirements.

Experimental results are obtained from the *QEyeTunnel* industrial automatic control quality system employed in the production line at the Mercedes-Benz factory in Vitoria, Spain. The paper describes briefly both the hardware and software architectures of this system, focusing on the industrial vision system in which the algorithm has been implemented. We provide the computational cost of our algorithm when it is implemented in a dedicated vision system in both *CPU* and *GPU* platforms. We also provide examples of the detection performance and analyze its robustness and reliability by carrying out several tests.

The paper is organized as follows: in Sec. 2, the present work objectives and assumptions are outlined. Then, in Sec.3, the deflectometry, image fusion and optical flow techniques are briefly described. Afterwards, the proposed approach is explained in detail in Sec. 4. Sec. 5 explains the experimental industrial setup together with all tests and results, complete with comments. Sec. 6 discusses some details of our proposal where improvement is possible. Finally, conclusions are provided in Sec. 7.

2. Objectives and assumptions

a) Objectives:

- Detection of defects cataloged as dings and dents (i.e., scratches, drop drains and all kind of surface deformations grater than $\varnothing 1\text{mm}$).
- Avoid the use of precomputed image background references.
- Detection in the entire illuminated area (i.e., in style lines).
- Accomplish with the assembly plants of passenger cars time production requirements (i.e., 30 seconds).

b) Assumptions:

- Use of gray scale cameras and light in the visible spectrum.
- The cameras are positioned and calibrated for technically accomplish with most automobile manufacturing standards.
- The inspected area is well illuminated.
- The illumination is based on projecting regular patterns sweeping the surface.

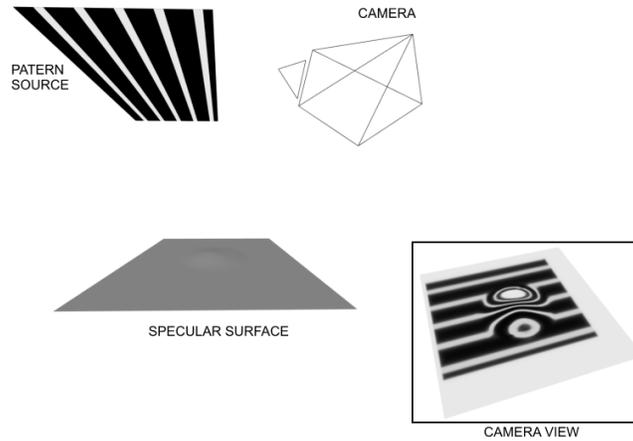
3. Materials and Methods

This section briefly reviews previous well-known results from literature that will be subsequently used by the detection method proposed in this paper: *deflectometry principle*, *image fusion* and *Lucas-Kanade optical flow*.

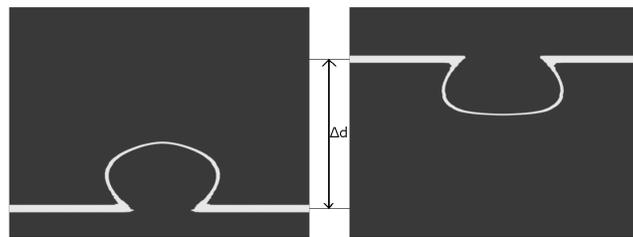
3.1. Deflectometry principle

This technique has proven to be a reliable and accurate approach to accomplish the task of detecting defects on car body surfaces [20]. In fact, this is what check-men do nowadays in manual inspections using their own eyes, analyzing the reflections reflected on the car body and looking for deformations. In the same way, as in the case with projection techniques, deflectometry is based on the projection of structured light patterns over a surface. When a triangulation method like fringe projection is used, the camera is focused on the surface onto which a light pattern is projected. Thus, when a deformation (i.e. ding or dent) appears, the light rays are deviated, producing a sudden change in the shape of the pattern, as shown in Fig. 3.

To the best of the authors knowledge, all current industrial inspection systems are based on this technique[10, 12, 21], as well as most of current research works in the literature. The main reason is that, by using the deflectometry principle followed by enhancement techniques, as the one proposed in this paper, it is possible to detect the majority of defects appeared in car body surfaces, independently their nature, and providing thus global solutions and feasible automatic industrial detection systems.



(a) Deformation of the projected pattern due to surface defects.



(b) Light behavior in two different frames when a dent is present.

Figure 3: Deflectometry based approach.

Indeed, and as it is commented in the Introduction section, we humans rely also on this principle to be able to notice any anomaly in these kind of surfaces. Said this, Fig. 4 shows two examples of raw images obtained by two different industrial systems that base the detection in the deflectometry principle.

3.2. Image fusion technique

The goal of image fusion is to integrate complementary multi-sensor, multi-temporal and/or multi-view information into one new image containing information about the quality of which cannot be achieved otherwise [22]. In this paper, the term refers the process in which several images acquired by one single camera at different instants of time ($k = 0, \delta, 2\delta, \dots$, being δ the camera's frame rate) are merged forming a new image (coined as fusion image), enhancing the interesting information for defect detection.

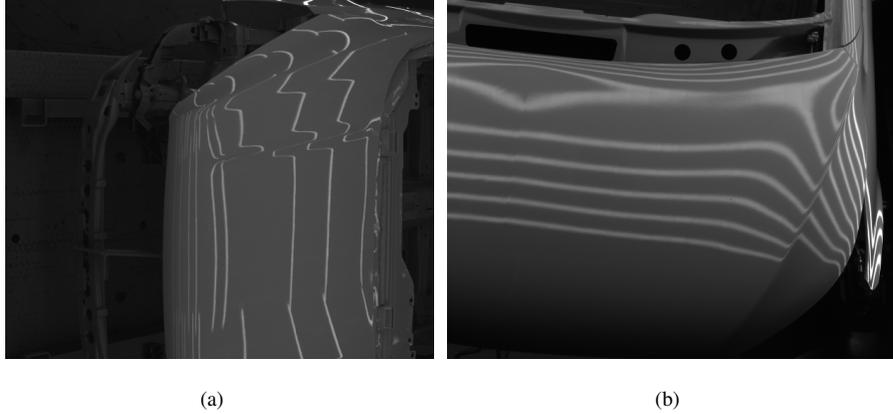


Figure 4: Raw images from DIP system (courtesy of Ford España S.A.U., Almussafes Plant) in (a), and *QEyeTunnel* system (courtesy of Mercedes-Benz España S.A.U., Vitoria Plant) in (b).

3.3. Lucas and Kanade optical flow estimation

Let $\mathbf{i}(\mathbf{x}) \in \mathbb{R}^{\mathbf{X}}$ be the acquired image at an instant of time t , in which \mathbf{X} represents the n -dimensional Euclidean space \mathbb{R}^l with $l = 1, 2$, or 3 together with the discrete topology, although other topologies such as *von Neuman* or the *odd-even product* are also commonly used in computer vision [23]. So, in this paper, $\mathbf{X} = \mathbb{Z}_m \times \mathbb{Z}_n = \{\mathbf{x} := (x, y)^T \in \mathbb{Z}^2 : 0 \leq x \leq m - 1, 0 \leq y \leq n - 1\}$, m and n being the number of columns and rows of the image. In addition, let $\mathbf{i}_x := \frac{\partial \mathbf{i}(\mathbf{x})}{\partial x}$, $\mathbf{i}_y := \frac{\partial \mathbf{i}(\mathbf{x})}{\partial y}$ and $\mathbf{i}_t := \frac{\partial \mathbf{i}(\mathbf{x})}{\partial t}$ denote the partial derivatives.

Optical flow is the apparent 2D motion in an image sequence, a $\mathbb{R}^3 \rightarrow \mathbb{R}^2$ flow-field in where $\mathbf{u} := (u, v)^T$ is called the *optic flow*. An important assumption is the *brightness constancy constraint* or *optical flow constraint* equation, [24]:

$$\mathbf{i}_t + u \cdot \mathbf{i}_x + v \cdot \mathbf{i}_y = 0 \quad (1)$$

The work presented by Lucas and Kanade in [25] is possibly one of the most studied approaches solving the optical flow constraint. The *LK-algorithm* consists of minimizing the following error:

$$E_{LK} = \frac{1}{2} \int \int \int_{\Omega} (\mathbf{i}_t + \mathbf{u}^T \cdot \nabla \cdot \mathbf{i})^2 \cdot d\mathbf{x} \quad (2)$$

yielding to the following result:

$$\mathbf{u}_{LK} = (\mathbf{G}^{2D})^{-1} \cdot \mathbf{b} \quad (3)$$

where $\mathbf{G}^{2D} = \begin{pmatrix} \int \int_{\Omega} \mathbf{i}_x^2 \cdot d\mathbf{x} + c & \int \int_{\Omega} \mathbf{i}_x \cdot \mathbf{i}_y \cdot d\mathbf{x} \\ \int \int_{\Omega} \mathbf{i}_x \cdot \mathbf{i}_y \cdot d\mathbf{x} & \int \int_{\Omega} \mathbf{i}_y^2 \cdot d\mathbf{x} + c \end{pmatrix} := \begin{pmatrix} m_{200} & m_{110} \\ m_{110} & m_{020} \end{pmatrix}$,
 $\mathbf{b} = \begin{pmatrix} \int \int_{\Omega} \mathbf{i}_x \cdot \mathbf{i}_t \cdot d\mathbf{x} \\ \int \int_{\Omega} \mathbf{i}_x \cdot \mathbf{i}_y \cdot d\mathbf{x} \end{pmatrix} := \begin{pmatrix} m_{101} \\ m_{110} \end{pmatrix}$, and c is the tunable Tikhonov constant to the m_{200} and m_{020} moments (further details in [26]).

The image flow is then obtained by developing Eq. 3, yielding to the following expression:

$$\mathbf{u}_{LK} = \begin{bmatrix} \frac{m_{011} \cdot m_{110} - (m_{020} + c) \cdot m_{101}}{-m_{110}^2 + (m_{020} + c) \cdot (m_{200} + c)}, & \frac{-(m_{011} \cdot m_{200} - m_{101} \cdot m_{110})}{-m_{110}^2 + (m_{020} + c) \cdot (m_{200} + c)} \end{bmatrix}^T \quad (4)$$

4. Proposed Approach

Fusion image techniques and deflectometry principle have been used before in the automotive industry to detect small defects (between $\varnothing 0.2\text{mm}$ and $\varnothing 1\text{mm}$) on specular surfaces, which are mostly produced by dirt on the surface when the car body is being painted². The majority of them rely on processing the image intensity maps [10, 14, 12, 6], since small defects tend to make the surface less specular or non-specular at all. The main problem is that defects cataloged as dings and dents are not detected by these approaches because the painting surface remains almost as specular as the rest of the non-defected surface (e.g., think in drops of paint). One example of this is given in Fig. 5, which shows the result of the DIP system algorithm [10] when a dent is present. Note that the defect can be hardly seen in Fig. 5(b).

In this paper, and contrary to current approaches, we propose to look at the velocity variations that a light pattern sweeping the surface suffers when an anomaly is present. Fig. 3(b) shows the effect of this variation and in the following we describe how to make use of this phenomenon to enhance dings and dents and detect them (see the result of our approach in Fig. 5(c)).

²For more details, visit <https://www.youtube.com/watch?v=HroEU8XsaTU>.



(a) Raw image. (b) Result using DIP [10]. (c) Result of our approach.

Figure 5: Example of a dent on the surface.

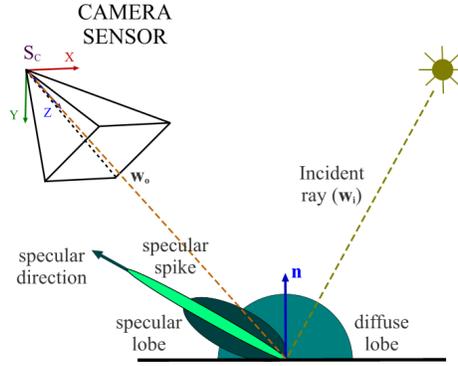


Figure 6: Reflectance model [27]. When light is incident to a surface, the light is reflected as a diffuse lobe, specular lobe and a specular spike, their shape and size dependent on the surface properties.

Let $\mathbf{f}_m \in \mathbb{R}^X$ and $\mathbf{f}_a \in \mathbb{R}^X$ be the resulting fusion images related to the magnitude and the angle of the optic flow, respectively. Therefore, our algorithm can be divided into two main steps based on the execution of the same:

- *Pre-processing step*, where the combination of N images acquired during the scanning stage is performed, obtaining as a result the fusion image/s:

$$\mathbf{f} = f(\mathbf{u}_{LK,k}(\mathbf{i}_k, \mathbf{i}_{k-1}), p_k(\boldsymbol{\omega}_i)) \quad (5)$$

where $\mathcal{P} = \{p_k(\boldsymbol{\omega}_i), k = 1, \dots, N - 1\}$ is the set of patterns reflected on the surface, p_k represents the intensity of the pattern at the location indicated by the incident ray $\boldsymbol{\omega}_i$ (see Fig. 6), and $k = 1, \dots, N - 1$.

- *Post-processing step*, where from the resulting fusion image/s, a discrimination

of the part of the image that is background and the part that is defect is performed by a treatment based on the local trend of each pixel followed by a threshold that makes it possible to distinguish between defect and noise. This treatment allows us to detect dings and dents in flat areas or with sudden changes in the surface such as style lines. As far as the authors are aware, this is the first work that proposes a solution which is able to detect dings and dents in such areas and that fulfills cycle time production line requirements.

It should be remarked that the *pre-processing step* must be carried out in real-time while the light pattern is sweeping over the body surface, ensuring that no frame is missed, which would produce a poor result. The *post-processing*, on the other hand, starts once the fusion image is ready, allowing us to perform other tasks simultaneously as long as the cycle time requirements are met.

In the following, we detail all the steps of the solution proposed in this paper, putting special emphasis on those aspects which differentiate it from other solutions.

4.1. Pre-processing step

Our image fusion proposal is based on two premises: first, to lose the minimum information possible in each of the merges; second, to meet the cycle time production line requirements. Let $\mathbf{u}_{LK,k} := (u_{LK,k}, v_{LK,k})^T$ denote the computed *optic flow*. Let $\mathbf{m}_k \in \mathbb{R}^X$ be the computed *optic flow magnitude* at instant k as:

$$\mathbf{m}_k = \sqrt{u_{LK,k}^2 + v_{LK,k}^2} \quad (6)$$

Similarly, let $\mathbf{a}_k \in \mathbb{R}^X$ be the computed *optic flow angle* at instant k , taking the following expression:

$$\mathbf{a}_k = \arctan\left(\frac{v_{w,k}}{u_{w,k}}\right) \quad (7)$$

Therefore, the fused magnitude image $\mathbf{f}_m \in \mathbb{R}^X$ is obtained using the computed *optic flow magnitude* as:

$$\mathbf{f}_m(i, j) = \{\mathbf{m}_1(i, j) \vee \mathbf{m}_2(i, j) \vee \dots \vee \mathbf{m}_{N-1}(i, j)\} \quad (8)$$

in where \vee represents the maximum operator between two values and (i, j) denoting the evaluated pixel of \mathbf{f}_m and \mathbf{m}_k .

At the same time, the fused angle image $\mathbf{f}_a \in \mathbb{R}^X$ is computed using both *optic flow angle* and *optic flow magnitude* as:

$$\mathbf{f}_a(i, j) = \begin{cases} \mathbf{a}_k(i, j) & \text{if } \mathbf{m}_k(i, j) > \mathbf{f}_m(i, j) \\ \mathbf{f}_a(i, j) & \text{otherwise} \end{cases} \quad (9)$$

$\forall k = \{1, 2, \dots, N-1\}$. Note that our proposal assumes that a big variation of the *optic flow magnitude*, due to a sudden change of the shape, is directly related with a big variation of the *optic flow angle*. Hence, the proposed Eqs. 8 and 9 will highlight all deformations existing on the surface.

Fig. 7 shows the performance of the presented approach in three consecutive instants: the first row, Figs. 7(a) to 7(c), shows the input images of the algorithm, which are the projection of the moving patterns on the surface; the second row, Figs. 7(d) to 7(f), shows how the fusion image of the optical flow magnitude \mathbf{f}_m is formed; the third row, Figs. 7(g) to 7(i), shows how the fusion image of the optical flow angle \mathbf{f}_a is formed³. It is remarkable that as the light sweeps the surface, our fusion technique enhance all changes in the same, making it clearer that a defect is present. Whilst our fusion strategy enhance defects, it also enhances areas which are not defects, such as corners or style lines. The *post-processing step* in our method provides a feasible solution to this problem, allowing the inspection and detection in areas with style lines.

4.2. Case Study: Non-deflectometry vs Deflectometry Techniques

In this section the benefits of using our proposed image fusion law for enhancing the anomalies on specular surfaces compared with other methods are shown. For the sake of readability, the following nomenclature is used: IWD, solution without using the deflectometry principle; KPLD, solution using the approach in the DIP system; and

³To see the complete sequence and more examples, visit <https://media.upv.es/player/?id=41c63310-5ed6-11e7-9b6d-8d718586122a>.

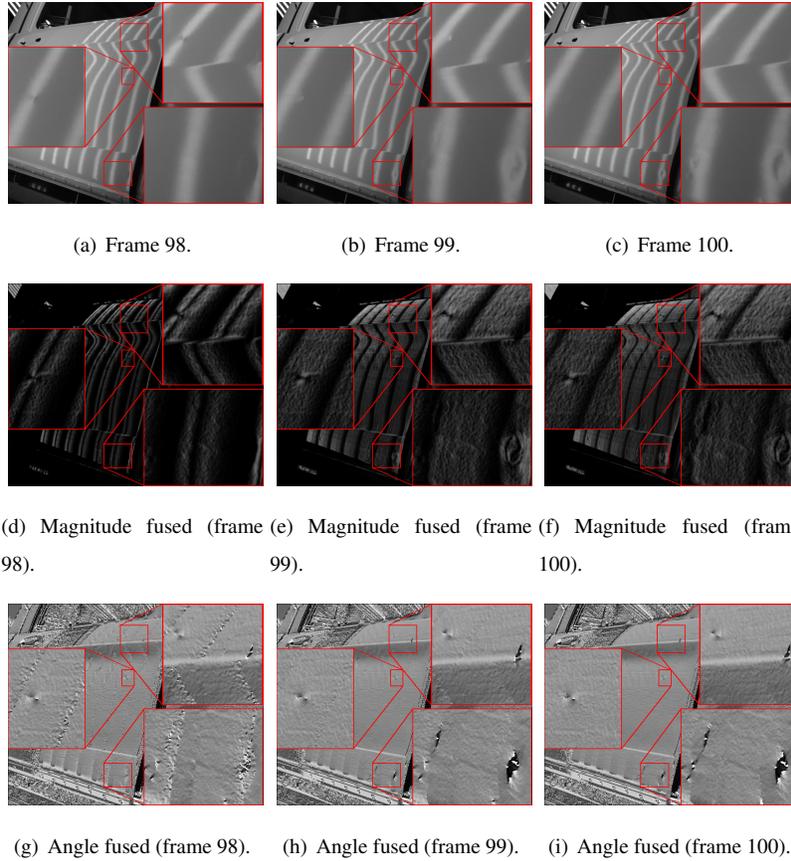


Figure 7: *Pre-processing step* algorithm performance.

OFD, solution using our approach⁴.

Fig. 8 shows the performance obtained by IWD (8(a), 8(b) and 8(c)), KPLD (8(d), 8(e) and 8(f)) and our OFD (8(g), 8(h) and 8(i)) when facing small dents. Note as without the use of deflectometry techniques, this kind of defects cannot be detected. With respect to the KPLD solution, note as the dent is masked by the image noise.

⁴Some industrial solutions claim to be able to detect also dings and dents, such as in [21, 28, 12]. However, there is no detailed information about the particular methodology/algorithm used, nor about their limitations (i.e. defect size, inspection coverage, etc.). If main details of such methods/algorithms, as well as setups and limitations are public in the future, it would be interesting to carry out some comparative studies, as an extension of the results presented in this paper.

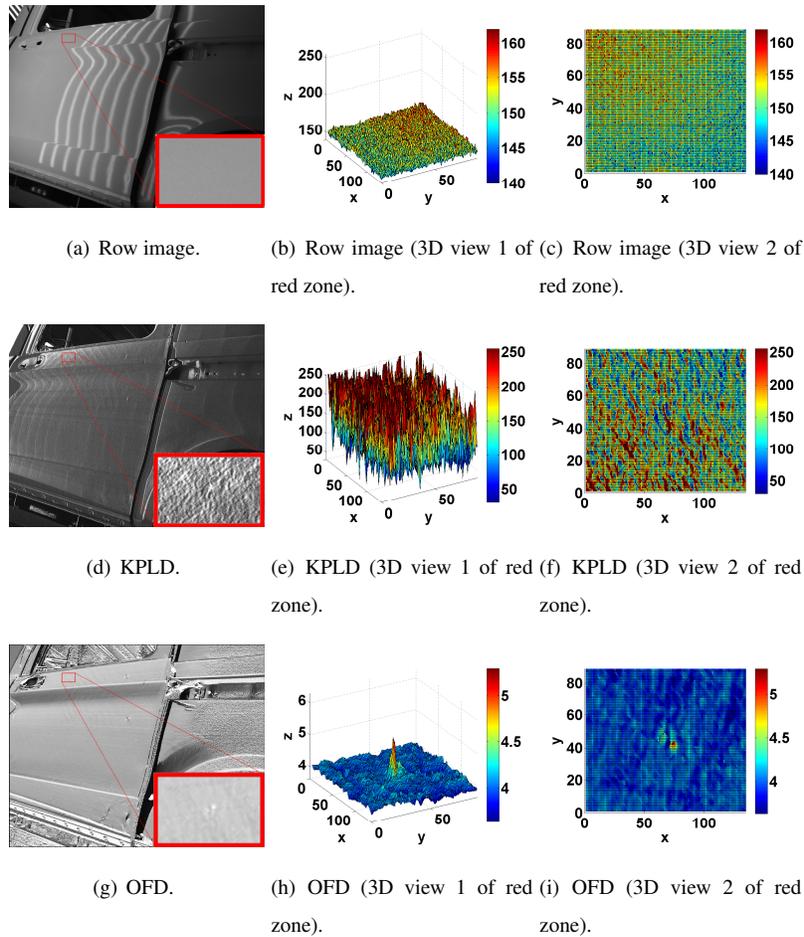


Figure 8: Performance of the studied approaches when facing small dents.

Contrary to this, our OFD enhance the present dent, making possible its detection.

In addition, Fig 9 shows the performance of KPLD and our OFD methods when facing a big ding. Even that in both approaches the defect could be identified, note that, in the case of KPLD the defect is barely distinguished from other parts of the image (see 9(a),9(b) and 9(c)), whilst OFD provides a much better solution where defect is highly enhanced and, as consequence, easier to be detected (see 9(d), 9(e) and 9(f))⁵.

⁵To see more examples, visit <https://media.upv.es/player/?id=fdc760c0-5e31-11e7-9b6d-8d718586122a>.

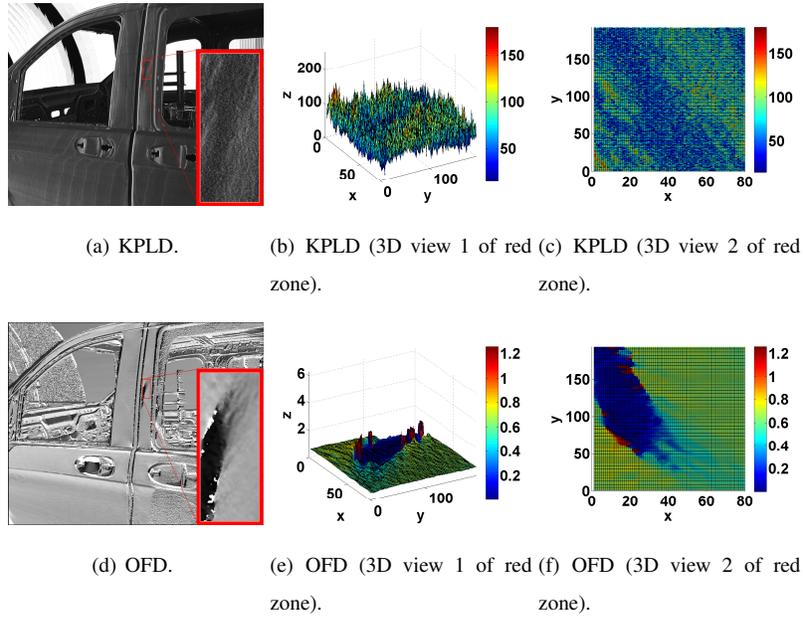


Figure 9: Performance of the studied approaches when facing big surface deformations.

4.3. Post-processing step

Unlike the majority of methods of defect detection that we can find in the literature [29, 30], which are based on obtaining a background of the analyzed surface off-line and then comparing it with the result of each scan, we propose a method that obtains the corresponding image background online, which is then used by the proposed defect detection algorithm, and meets the cycle time required by the production line. As far as we are aware, this is the first work detailing a method able to deal with smooth changes of the surface such as style lines, for dings and dents detection on specular surfaces. Until the present day, such parts were excluded/hidden, reducing the detection area considerably. Note that for doing this, a perfect calibration of the vision system must be performed in addition to a $2D$ or $3D$ matching of the surface seen by each camera. This usually requires the use of well-known patterns that are not always found in sufficient numbers in order to achieve a correct match. In addition, such techniques significantly increase the computation time of the algorithms [31].

Therefore, our approach for detecting defects is based on the sudden changes of

direction that these defects produce in the light pattern projected on the surface. Hence, only the information contained in \mathbf{f}_a is used. Instead of working directly with angles, to avoid some computational issues we split \mathbf{f}_a in the following two matrices:

$$\begin{aligned}\mathbf{g}_x(i, j) &= \cos(\mathbf{f}_a(i, j)) \\ \mathbf{g}_y(i, j) &= \sin(\mathbf{f}_a(i, j))\end{aligned}\tag{10}$$

where $\mathbf{g}_x \in \mathbb{R}^{\mathbf{X}}$ represents the contribution of the optical flow in the x -direction, $\mathbf{g}_y \in \mathbb{R}^{\mathbf{X}}$ its contribution in the y -direction, and (i, j) denoting the evaluated pixel. For the sake of readability, since the same method is applied to both \mathbf{g}_x and \mathbf{g}_y matrices, from now until further notice let $\mathbf{g}_l \in \mathbb{R}^{\mathbf{X}}$ represent any of these matrices, with $l = \{x, y\}$.

Note that, the basis of the our image background subtraction method is based on obtaining the correct directions (trend) of each pixel. These directions are obtained from the estimated pixel flow. In order to avoid bad pixel directions produced by noise or bad computations and to get correct directions, we propose smoothing them in accordance with the trend of their neighborhood. So first a linear filter, such as a box filter, is applied. Let $\mathbf{d} \in \mathbb{R}^{\mathbf{X}}$ be the resulting smoothed directions of \mathbf{g}_l :

$$\mathbf{d}_l := \mathbf{g}_l \left(\Gamma \right) N \tag{11}$$

where $\Gamma : \mathbb{R}^{\mathbf{X}}|_N \rightarrow \mathbb{R}$ denotes the averaging function $\Gamma(\mathbf{g}_l|_{N(\mathbf{y})}) = \frac{1}{\text{card}(N(\mathbf{y}))} \sum_{\mathbf{x} \in N(\mathbf{y})} \mathbf{g}_l$, and $N(\mathbf{y}) \subset \mathbb{Z}^2$ a predefined neighborhood of $\mathbf{y} \in \mathbb{Z}^2$.

Therefore, the image background is obtained as follows. Let \mathbf{h} be a subset of \mathbf{d}_l with radius R . For the sake of formulation simplicity, let $c = (i, j)$ be the reference pixel and $p = (-R \leq i \leq R, -R \leq j \leq R)$ as the evaluated pixel. Let \mathbf{h}_c be the center of the subset, and $\mathbf{v} := \mathbf{d}_c$ its vector director. For every pixel \mathbf{h}_p of the subset, we compute the vector $\mathbf{z}_p = \mathbf{h}_p - \mathbf{h}_c$, taking the center cell of the subset as the reference cell (see Fig. 10). Hence, its projection over the direction given by \mathbf{h}_c is:

$$d_p = |\mathbf{v} \times \mathbf{z}_p| \tag{12}$$

With this, we define a weighting value for each pixel of the subset in the form:

$$w_p = (0 \vee (R - d_p))^3 \tag{13}$$

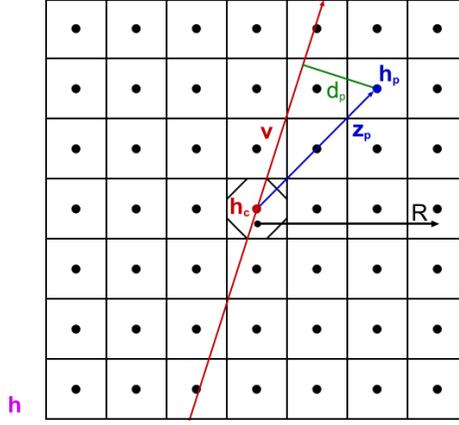


Figure 10: Proposed local directional-based blurring method.

Note that, with this weighting, pixels of the subset whose directions are close to the direction of \mathbf{h}_c (small d_p) have greater weight, while pixels whose directions are not related to the direction of \mathbf{h}_c (large d_p) have less weight. Also note that, for pixels where $(R - d_p) < 0$, $w_p = 0$, there is no blurring⁶.

Hence, the background image is given by:

$$\mathbf{s}_l = \frac{\sum (w_p \cdot \mathbf{d}_{l,p})}{\sum w_p} \quad (14)$$

Hence, once the background image is obtained, the next step is to differentiate defects from noise. Let denote $\mathbf{s}_x \in \mathbb{R}^{\mathbf{X}}$ as the image background of \mathbf{g}_x and $\mathbf{s}_y \in \mathbb{R}^{\mathbf{X}}$ the image background of \mathbf{g}_y . In addition, let $\mathbf{t} \in \{-1, 1\}^{\mathbf{X}}$ be the resulting matrix computed as follows:

$$\mathbf{t}(i, j) = \frac{\mathbf{g}_x(i, j) \cdot \mathbf{s}_x(i, j) + \mathbf{g}_y(i, j) \cdot \mathbf{s}_y(i, j)}{\|\mathbf{g}_x(i, j), \mathbf{g}_y(i, j)\|_2 \cdot \|\mathbf{s}_x(i, j), \mathbf{s}_y(i, j)\|_2} \quad (15)$$

where $\|(x, y)^T\|_2 = \sqrt{x^2 + y^2}$ is the *Euclidean norm*. Note as the result shown in Eq. 15 will highlight the differences that \mathbf{g}_x and \mathbf{g}_y have with \mathbf{s}_x and \mathbf{s}_y , respectively.

⁶Other possibility is to use the well-known Gaussian weighting function [32]. Studies of image similarity indicated that there was no significant difference between both solutions (i.e., the structural similarity index SSIM=0.984). However, the use of the Gaussian weighting function increased almost 1.7 times the computational cost of the proposed algorithm.

In our particular case, the normalization term $\|\mathbf{g}_x(i, j), \mathbf{g}_y(i, j)\|_2 := 1 \forall (i, j) \in \mathbf{X}$ since the vector $(\mathbf{g}_y(i, j), \mathbf{g}_y(i, j))^T$ is unitary $\forall (i, j) \in \mathbf{X}$ (see Eq. 10).

The last step of the algorithm consists of performing a *global thresholding* of the resulting image \mathbf{t} , providing thus the result of the detection. Hence, let $\mathbf{r} \in \{0, 1\}^{\mathbf{X}}$ be the image obtained after the commented operation:

$$\mathbf{r} = \chi(\mathbf{t}) \quad (16)$$

where $\chi = \begin{cases} 1 & \text{if } \mathbf{t} < th \\ 0 & \text{if } \mathbf{t} \geq th \end{cases}$ and th the tunable threshold value. The method for obtaining threshold values automatically is beyond the scope of this paper.

Finally, a masking layer is applied to all those areas seen by the camera that either do not belong to the surface of the body (e.g. mechanical or production line structures), or parts in which detection is not desirable (e.g., interior of the window frame or holes in the surface). The method for generating image masks, which is very specific to the body shape and system structure, is beyond the scope of this paper. Let $\mathbf{n} \in \{0, 1\}^{\mathbf{X}}$ be the correspondent mask image. Thus, the result $\mathbf{r}_n \in \{0, 1\}^{\mathbf{X}}$ is:

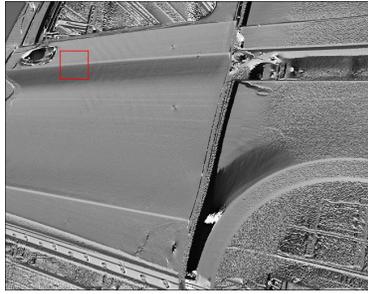
$$\mathbf{r}_n := \mathbf{r} \& \mathbf{n} \quad (17)$$

where the operator $\&$ represents the binary “AND” image operation.

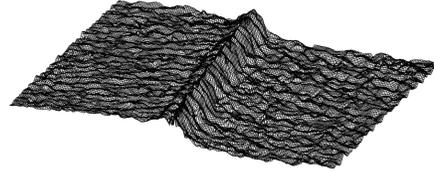
4.4. Case Study: Mean Filter vs Local Directional-based Blurring

In this section the benefits of using our local directional-based blurring method as image background subtraction are shown. Note that our approach is based on blurring the original image according to the pixel trend. Other possible and easiest method is using a filter in order to blur the image, such as the mean filter. In this particular case, the method replaces each pixel value in an image with the average value of its neighbors, including itself.

Let \mathbf{f}_a be the fusion image shown in Fig. 11(a). Fig. 11(b) shows the 3D representation of the area framed in red. Note in the figure how the style line is clearly identified.



(a) Fusion image f_a .



(b) 3D representation of the zone highlighted in f_a .



(c) Background solution given by the mean filter (box size 10).



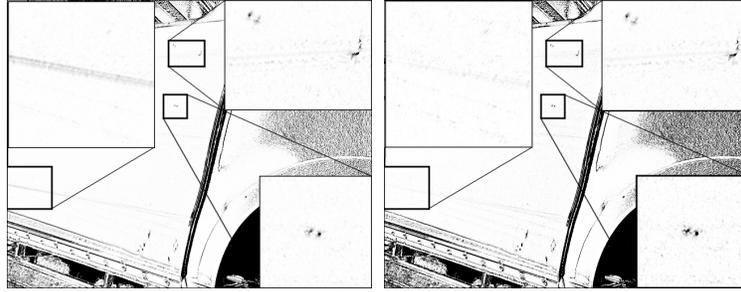
(d) Background solution given by the local directional-based blurring method (box size 10).

Figure 11: Case study: comparison of the solution given by the mean filter and the local directional-based blurring methods.

Using a mean filter with a box of size 10 pixels, the background takes the form shown in Fig. Fig. 11(c). Note that the image noise has been reduced but the style line remains, although differs considerably to the original in Fig. 11(b).

Likewise, using the method detailed from Eq. 11 to 14 with a box of size 10 pixel, the background takes the form shown in Fig. 11(d). Note how the image noise has been reduced as when using the mean filter, but the style line remains almost the same like the original in Fig. 11(b), which is one of the biggest advantages put forward in our approach.

Thus, when we use Eq. 15 using the background obtained with the mean filter the result is the one shown in Fig. 12(a), whilst using the proposed local directional-based blurring method the result is the one shown in Fig. 12(b). Both images shows in plane areas almost the same result but in areas with style lines, these are present when using the mean filter approach and are almost disappeared when using the local directional-based blurring method.



(a) Background computed using a mean filter with box size 10. (b) Background computed using the local directional-based blurring method (box size 10).

Figure 12: Resulting image t : a) using the mean filter; b) using the local directional-based blurring method.

5. Experimental Evaluation and Results

As seen in Sec. 4, the defect detection algorithm proposed in this paper consists of two phases: a *pre-processing* step, in which a light scan of the body to be analyzed is required using light patterns (e.g., straight lines, chess board pattern, etc.), and a *post-processing* step, which uses as input the fusion image resulting from the scan performed in the previous phase. For the first phase, it is necessary to use some hardware that projects light patterns and performs this scanning process. Although a laboratory setup could have been used for validation, we have taken results from the *QEyeTunnel* industrial system recently developed in Mercedes-Benz' factory in Vitoria, Spain (Fig. 13). Following is a brief description of this system avoiding all industrial aspects and those related to the integration of the system. We have also omitted other aspects such as camera positions, intrinsic parameters, light patterns, etc., which are not deemed important in this paper.

5.1. *QEyeTunnel: an Automated Vision-based Quality Control Inspection System*

In general terms, this inspection system consists of two parts: an external fixed structure where a determined number of cameras are optimally placed in order to see the entire surface of the body to be analyzed, as well as a moving internal structure similar to a scanning machine which houses curved screens known as 'sectors' which



Figure 13: *QEye-Tunnel*: an Automated inspection system for surface body-car quality control based on artificial vision.

act as the light sources that project the illumination patterns over the body surface, as can be seen in Fig. 13. The number of cameras as well as the number and size of sectors depend on the car bodies to be analyzed by the system. In particular, the *QEyeTunnel* inspection system employed at Mercedes-Benz in Vitoria comprises 23 monochrome cameras of 5MP working at 15fps, and two sectors set 3 meters apart, with a resolution of 192×1344 each and a total route displacement of 2m moving at 0.267m/s of speed. With this setup, the scanning stage takes around 11s from start to finish.

Fig. 14 shows the elements of the hardware architecture as well as the communication relationships in the *QEyeTunnel* system. The main features are:

- *System Low-level Controller (PLC)*: This controls the mechanical movement of the sectors and also monitors all production line stages, system security as well as visual and sound alarms. The PLC is connected using the *PROFISAFE* communication protocol, and to the *PC PROC* using the *TCP-IP* protocol, through the use of a *PROFISAFE-ETHERNET* converter. The characteristics of each body as well as its position during the inspection process is indicated by this *PLC* to the *PC PROC*. The *PLC* waits for the starting signal from the *PC PROC* before sending back the ending signal when the scanning stage has finished.

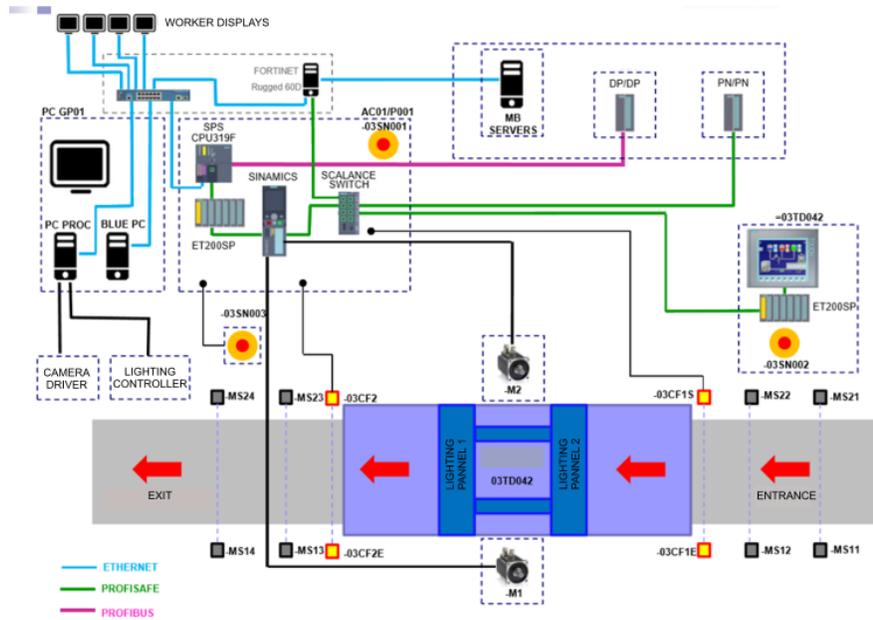


Figure 14: Hardware architecture and communication relationships of the *QEyeTunnel* system.

- *Vision Industrial System (PC PROC):* This is the element that controls the program flow. It is an industrial vision system from *Matrox*[®] named *Supersight Solo*, which is an entry-level configurable single-node high-performance computing (HPC) platform supporting two multi-core *Intel*[®] *Xeon* processors, third-party *GPUs* and *Matrox*[®] *FPGA* boards for demanding industrial imaging applications. It is equipped with 6 *GigE Card PCIe AdLink*[®] *GIE64+* with Power over Ethernet technology (*PoE*), and 4 ports for supporting 24 cameras. Moreover, it is equipped with 2 *Gigabyte*[®] *GeForce GTX 1080 8GB GDDR5X Dual Link DVI-D HDMI 3X DisplayPort PCI-E* graphic cards, which were used for the implementation of our proposal and also to run the program which generates the patterns to be projected. This system communicates with the *PLC* and the *BLUE PC* using the *TCP-IP* communication protocol.
- *System High-level Controller (BLUE PC):* This is responsible for providing the inspection results to the production line workers, as well as acting as an interface

between line workers and the *PC PROC* on maintenance issues. It is an industrial *PC*-based system which communicates with the various *Worker Displays*, *MB Servers* and *PC PROC* systems using the *TPC-IP* communication protocol.

- *Worker displays*: Several screens controlled by small *PCs* are situated throughout the production line to display the results of the inspection. Workers use the information to locate the defects and act accordingly (fixing them when possible or marking them for later repair). Substituting such screens with light-weight augmented reality glasses is currently being considered, which would improve worker comfort and facilitate the identification and repair processes.
- *MB Servers*: The results of each inspection as well as the system backups are saved on these servers. Communication with the *BLUE PC* is through *TCP-IP* but using a *FORTINET* protocol. Apart from serving as a data security system, it is also used to perform big-data analysis in order to identify problems in the painting process, which will ultimately result in significant money savings and an increase in the quality of the final product.

Fig. 15 shows the inspection program flow as each car body enters the system. When a new body is ready to be introduced into the inspection system, the production line control systems inform the *PLC* accordingly and also provide all necessary body-related data (e.g. model, color, etc.). Only the necessary data is transmitted by the *PLC* to the *PC PROC* which, while the body is being positioned correctly for inspection, loads all initialization parameters, such as thresholds, masks, light pattern, etc. Once the car body is in the inspection position (signal transmitted by the *PLC* to the *PC PROC*), the *PC PROC* gives scanning permission to the *PLC* and activates the cameras. The 23 cameras then begin to send images at 15fps to the *PC PROC* and both sectors begin the scanning movement as instructed by the *PLC*. The *PC PROC* computes the previously mentioned *pre-processing* step and at the same time waits for the *PLC* to send the scanning ending signal. With the end of the scanning, the *PLC* warns the *PC PROC* and the latter stops the acquisition of images and proceeds to perform the *post-processing* step for all cameras. Meanwhile, the *PLC* allows the car body to leave the

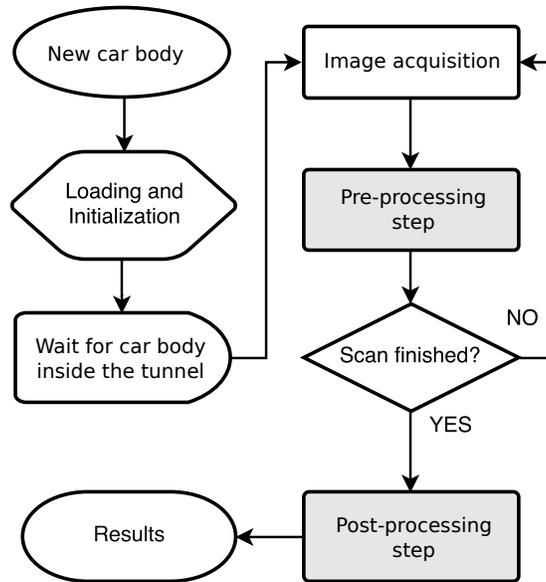


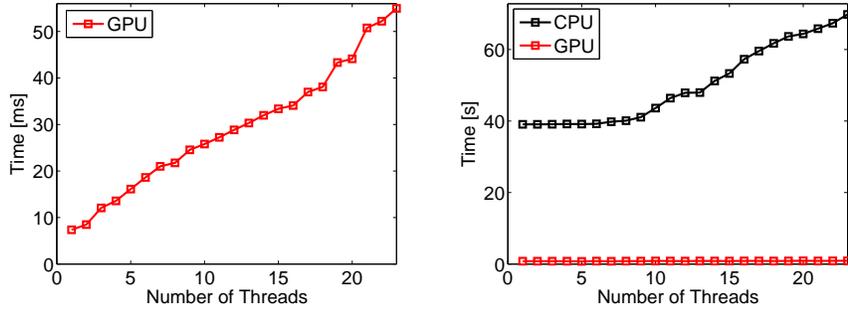
Figure 15: Inspection system program flow.

inspection system and, when a new body is ready, repeats the process. When the results of the inspection are obtained, the *PC PROC* sends them to the *BLUE PC*, which in turn sends them to the different display monitors so that workers can locate identified defects on the car body and see to their repair.

5.2. Algorithm performance

As mentioned in previous sections, this entire procedure, including the time required to place the car body in the inspection position and to return it to the production line, must not exceed the cycle time requirements of the production line. This time, in the case of passenger automobiles, is around $30s$. Taking into account that the scanning process takes about $11s$ and that the positioning of the vehicle and its return to the production line takes about $8s$, this usually leaves approximately $10s$ in which to carry out the entire detection process.

In view of the large amount of data to be processed and the short time to carry out this processing, we make use of *GPU-accelerated computing* [33], which refers to the use of graphics processing unit *GPU* together with a *CPU* to accelerate applications.



(a) *Pre-processing* step: maximum time per frame. (b) *Post-processing* step: maximum time.

Figure 16: Computational cost comparison between the proposed algorithm implemented in *CPU*⁴ and in *GPU*⁵. Performance given 10 iterations in each thread.

GPU-accelerated computing offloads compute-intensive portions of the application to the *GPU*, while the remainder of the code still runs on the *CPU*. *GPU-accelerated computing* together with the enormous power of current *GPUs* (in terms of computing power, energy consumption and heat dissipation) play a huge role in accelerating applications ranging from artificial intelligence to image processing, and robots.

In light of the previously mentioned, Fig. 16 shows the results of the computational cost comparison of the algorithm proposed in this paper optimally programmed in *CPU*- and *GPU*-based architectures, and using the previously mentioned *Supersight Solo* industrial vision system and the 23 cameras used by Mercedes-Benz' *QEyeTunnel* inspection system in Vitoria, Spain. It should be mentioned that in this analysis, we have excluded all computational costs and times required by the communication between the *PLC* and *PC PROC*, as well as the time needed to send the inspection results to the *BLUE PC*.

⁴The tests were performed using 2 Intel® Xeon® E5645 48GB DDR3-1066 processors. For more information about the main characteristics of the CPU, visit http://www.matrox.com/imaging/media/pdf/products/supersight/supersight_e2.pdf

⁵The tests were performed using 2 Nvidia® GeForce GTX 1080 (GP104-400) GPUs with 8GB GDDR5X of VRAM capacity. For more information about the main characteristics of the GPU, visit <http://www.nvidia.es/graphics-cards/geforce/pascal/gtx-1080/>

Thus, Fig. 16(a) shows the *pre-processing* step performance. First, *CPUs* are unable to handle and correctly perform the online processing required by the 23 cameras, and are not even able to support 1 camera (362.021 ms/frame). This is not only due to a computational power problem but rather to a RAM bus access problem. When this happens, the whole system freezes, resulting in a loss of frames, which is of course totally unacceptable. On the other hand, it is appreciated that the *GPUs* are able to support all cameras without problems, with the maximum time needed to run our algorithm being 55ms per frame.

With regard to the *post-processing* step, Fig. 16(b) shows the results obtained by both implementations, with 67s needed for the *CPU* implementation and less than 1s for the *GPUs* when dealing with 23 cameras. Again, using the *CPU* implementation it is not possible to accomplish the cycle time restriction of the production line, whilst in the case of the *GPUs* implementation it is possible to add additional detection algorithms or other kinds of computation such as the communications mentioned previously between the *PLC* and *BLUE PC*.

A simple way to understand the difference between the resulting *CPU* and a *GPU* performances in Fig. 16 is to compare how they process tasks. A *CPU* consists of a few cores optimized for sequential serial processing while a *GPU* has a massively parallel architecture consisting of thousands of smaller, more efficient cores designed for handling multiple tasks simultaneously. With respect to the memory access, *CPUs* architectures are designed to do huge amounts of computations and thus reduce the number of RAM accesses (with a bandwidth of around 15GB/s) to speed up the processing. On the other hand, *GPUs* architectures are designed to carry out few computations and lots of memory accesses (in this case VRAM with a bandwidth of around 300GB/s) in order to speed up parallel tasks. Image processing requires relative few computations and lots of memory accesses, which explains the improvement shown in the figure.

This study shows that the use of the *GPUs* not only allows our approach to be implemented in a system with so many cameras working in parallel but also allows the addition of more algorithms to detect other types of defects not identified by the algorithm proposed in this paper, possibly large dings and dents. That is why Mercedes-Benz' *QEyeTunnel* inspection system in Vitoria uses *GPUs* as the platform for processing

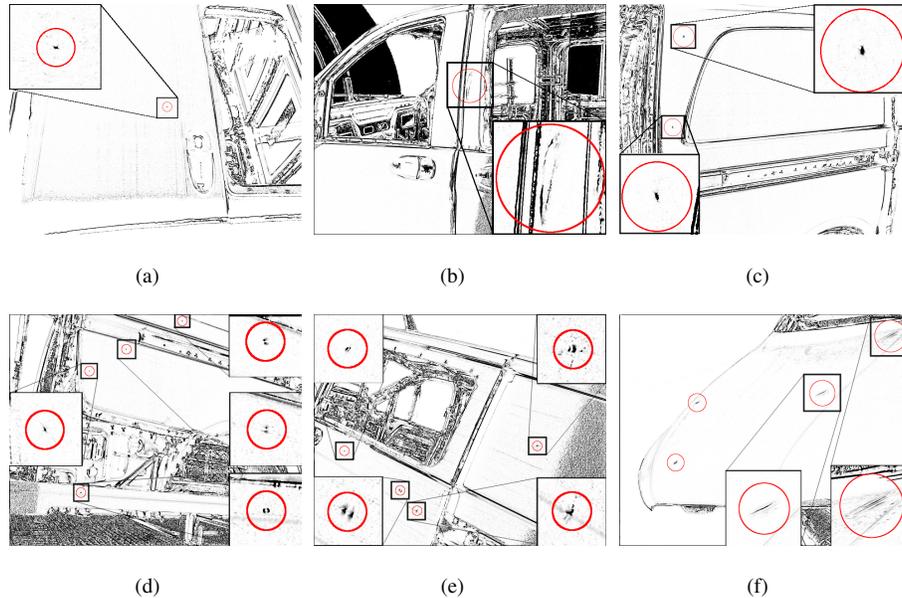


Figure 17: Set of detection examples showing different views and the detection of defects on different car bodies.

artificial vision algorithms and all results referred to here were obtained using implementation on these *GPUs*.

5.3. Defect Detection Analysis and Results

To demonstrate the performance of the defect detection algorithm presented in this paper, a selection of results obtained on the production line with different parts and car bodies is shown in Fig. 17. In some of these, detections are observed close to style lines, as in Figs. 17(a) and 17(b). In addition, it is possible to see detections carried out in parts where there is little surface area for detection such as the side door, Fig. 17(b). Note that if style lines have to be masked, giving in addition a margin of safety, as one can see, the area would be considerably reduced.

An interesting aspect to be analyzed is how external aspects such as the sectors and movement of illumination patterns, as well as varying car body positions while stationary in the inspection position, affect the proposed detection algorithm. For this, a car body with commonly found defects on the entire surface is used and two kinds of

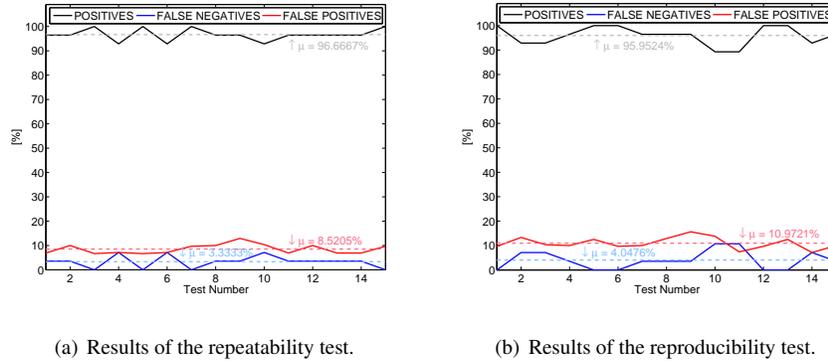


Figure 18: Results of the repeatability and reproducibility test

experiments are carried out:

- 1) The car body is placed in the inspection position and a total of 15 inspections are carried out without changing its position. The information obtained from this analysis will determine the variability in the detection taking into account only the variability of the movement of the light pattern and its reflection on the body. This analysis is known as a repeatability test.
- 2) The car body is introduced into the system and positioned in the inspection position. Once the inspection is completed, the car body comes out and the cycle is repeated 15 times. This will determine the effect on our proposal of the variability caused by the position of the car body, as well as other factors such as the movement of the same on the production line which can cause movement of some parts (i.e. side doors, rear doors or back door). This analysis is known as a reproducibility test.

Thus, Fig. 18(a) shows the results obtained in the first test, the repeatability test. It is observed that the detection has a variability of around 3.33%, which in terms of repeatability means we are at around 96.66%, a very satisfactory result in terms of industrial vision systems, thus corroborating the robustness of our proposal.

With regard to the reproducibility test, Fig. 18(b) shows the results obtained. These indicate that our proposal continues to have a variability of around 4%, which in terms

of reproducibility means we achieve around 96%, corroborating again the robustness of the proposal.

It should be noted that this study has indicated that most false positives (around 10%, as shown in the figures) are produced by the masks used to hide non desirable detected areas, such as holes, door windows, etc.

Finally, to give an indication of its detection capabilities, our algorithm is currently capable of detecting the following types of defects: dings, dents, runs, sanding scratches, moisture blisters of various types.

6. Discussion

A result to be discussed is the image fusion proposal shown in Eqs. 8 and 9, which uses the maximum operator. Although it has proved to be a good and feasible solution to the problem of detecting dings and dents, as it is shown along the paper and also with the algorithm working in a factory, nowadays we are only using the information regarding to the frame in which for a particular pixel the magnitude has been maximum, losing the information given by the rest of frames. Given that we think some of the information can be lost because of this, currently we are considering other merging options to lose as minimum information as possible during the fusion process and, hence, increase the detection capability of the presented algorithm.

Other aspect we like to discuss here is that the proposed *post-processing step* algorithm only uses the information of the fused angle image, \mathbf{f}_a , leaving unused the information of the magnitude fused image, \mathbf{f}_m . The main reason is the noise contained in \mathbf{f}_m , mostly produced by the orange peel of the surface, which is generated during the painting process. This level of orange peel is allowed by the manufacturer and it must not be detected. Fig. 19 shows an example of the magnitude and angle fusion images, Figs. 19(a) and 19(b) respectively. The reduction of the noise accomplishing the cycle time restriction is one of the other aspects in which we are currently working.

As regards the threshold values in Eq. 16, they are currently set manually following the next process: given a wide set of fusion images from different car bodies, regardless of the color, th is set so that false positives are minimized and true positives are max-

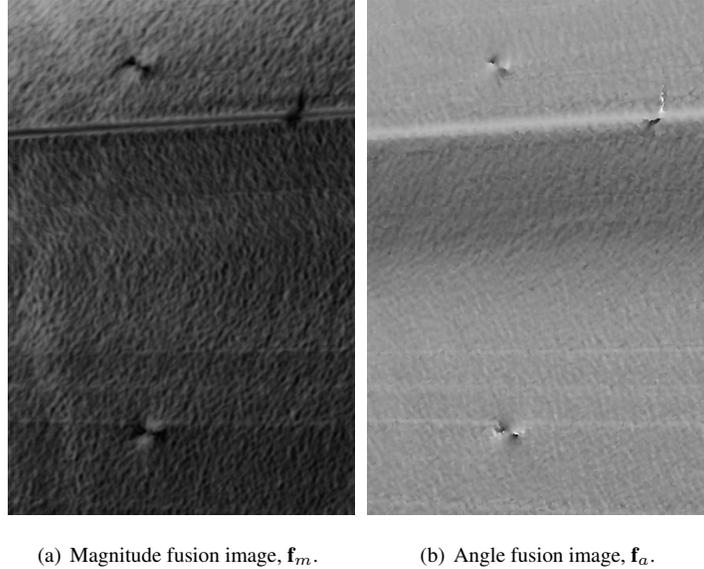


Figure 19: Noise effect.

imized. The optimal " th " is taken as the best for all the samples of the data-set. Since the knowledge of what is a true positive and what is a false negative is unknown a priori, a first adjust is carried out, which depends on the subjectivity and knowledge of the person in charge of the setting-up. A second adjustment is performed at the same time the system is in production, using the feedback from check-men. Due to each camera has a different threshold value, this method is repeated for each camera of the system. Given that this manual process is very time consuming in terms of human resources, nowadays we are working in a method for computing threshold values automatically.

Finally, it is worthy to discussed about the managerial implications of the proposed detection method. Although the method described in this work is generic, it is dependent of a mechanical system with cameras and light projectors. Hence, structural changes in current assembly plants would be needed in order to incorporate automatic inspection systems such as [14, 12, 21, 3].

Another implication is the modification of current working methodologies. As commented before, in most of the assembly plants the experts inspect the entire body surface, manually marking the defects for later repair. This methodology could be

change as follows:

- Results of the automatic inspection would be provided to the experts so that the latter would determine whether the defect would be repaired or not. With this proposal, the experts would avoid inspecting the entire body but still had the final decision. This could be carried out through screens located on the board [14, 21] or tablets managed by the experts.
- Results of the automatic inspection would be provided to the experts who would proceed to their repair. Manual marking for locating the defects before being repair could be done, such as in [14, 21], or automatic marking using, for instance, augmented reality.
- Results of the inspection would be provided to robotic systems that would proceed to their repair.

Lastly, “big-data analysis” techniques could be utilized to detect problems and malfunctions in previous stages since the result of each inspection could be easily storage. To be able to do this, it could be developed information storage and software tools that would imply to adapt or modify the existing information management of the assembly plant.

7. Conclusion

In this paper, a new vision algorithm based on deflectometry techniques for detecting dings and dents on specular surfaces in general, and car body surfaces in particular, has been presented. Our approach is based on the information obtained by reflecting a light pattern on the specular surface to detect structural deformations thereon.

The algorithm relies on two steps: a *pre-processing* step, performed during the scanning process, and a *post-processing* step, where, using the information from the *pre-processing* step, defects are detected.

With regard to the *pre-processing* step, we have detailed a new image fusion algorithm based on optical flow method that allows us to enhance the light variations

produced by sudden changes of the light while sweeping the surface when deformations appear, which indicates the presence of defects, style lines, edges, concavities, etc., and facilitating thus the defect detection in the later phase of the presented algorithm. As result, two fusion images containing different information are obtained: magnitude fused image and the angle fused image. In this work, only the angle fused image has been used.

As regards the *post-processing* step, a novel image processing is presented, which allows defect detection not only in flat surfaces but in those with sudden changes such as style lines, provided that the area is well lit. As far as we are aware, this is the first work dealing with this problem. Our approach uses a background subtraction technique based on a local directional-based blurring method through which low-level information in the angle fused image, showing large surface deformations such as style lines, is differentiated from high-level information, in which defects are found. Once the background is extracted, the dot product between the fused angle and background images has been used to highlight the discrepancies between both (defects). Finally, a global threshold is set in order to distinguish defects from noise. We have seen a typical example of a defect on a style line, and explained step by step the solutions given by the presented algorithm.

In addition, the paper shows and analyzes the results of the proposed detection algorithm implemented in the industrial *QEyeTunnel* inspection system developed at the Mercedes-Benz factory in Vitoria, Spain. First, the study of the algorithm's computational cost has shown that, in order to meet cycle time constraints of the production line, our algorithm needed to be implemented using a platform of *GPUs*, with a maximum processing time of *55ms* per frame in the *pre-processing* step and of less than *1s* in total in the *post-processing* step using 23 cameras working in parallel. Subsequently, several examples of defects located in or close to style lines, edges or corners have been shown, thus demonstrating the detection power of our algorithm. Finally, an analysis of the robustness of the proposed algorithm has also been carried out. The results have shown that the system has a variability of less than 4%, reporting an 11% of false positives.

Acknowledgments

This work is supported by VALi+d (APOSTD/2016/044) and PROMETEO (PROMETEOII/2014/044) Programs, both from Conselleria d'Educació, Generalitat Valenciana.

The authors would also like to thank the heads of project management and the check-men at the Mercedes-Benz factory in Vitoria, Spain, for all their efforts during the project, especially Juan Carlos Villanueva, Justo Oria and Juan Carlos Sadaba.

References

- [1] J. Pfund, N. Lindlein, J. Schwider, R. Burow, T. Blumel, K. E. Elssner, Absolute sphericity measurement: a comparative study on the use of interferometry and a shack-hartmann sensor, in: *CLEO/Europe Conference on Lasers and Electro-Optics*, 1998, pp. 201–201. doi:10.1109/CLEOE.1998.1115498.
- [2] R. Ritter, R. Hahn, Contribution to analysis of the reflection grating method, *Optics and Lasers in Engineering* 4 (1) (1983) 13–24. doi:http://dx.doi.org/10.1016/0143-8166(83)90003-9. URL <http://www.sciencedirect.com/science/article/pii/0143816683900039>
- [3] S. H'ofer, J. Burke, M. Heizmann, Infrared deflectometry for the inspection of diffusely specular surfaces, *Advanced Optical Technologies* 5 (2016) 377 – 387. doi:doi:10.1515/aot-2016-0051.
- [4] A. Yogeswaran, P. Payeur, Features extraction from point clouds for automated detection of deformations on automotive body parts, in: *2009 IEEE International Workshop on Robotic and Sensors Environments*, 2009, pp. 122–127. doi:10.1109/ROSE.2009.5355976.
- [5] V. Borsu, A. Yogeswaran, P. Payeur, Automated surface deformations detection and marking on automotive body panels, in: *2010 IEEE International Conference on Automation Science and Engineering*, 2010, pp. 551–556. doi:10.1109/COASE.2010.5584643.

- [6] Z. Sárosi, W. Knapp, A. Kunz, K. Wegener, Detection of surface defects on sheet metal parts by using one-shot deflectometry in the infrared range, ETH Zurich, IWF, 2010.
URL <https://books.google.com.au/books?id=8-w6AwEACAAJ>
- [7] J. Li, An Intelligent System for the Defect Inspection of Specular Painted Ceramic Tiles, University of Kentucky Libraries, 2005.
URL <https://books.google.com.au/books?id=bf0FkAEACAAJ>
- [8] N. Sameer Ahamad, J. Bhaskara Rao, Analysis and Detection of Surface Defects in Ceramic Tile Using Image Processing Techniques, Springer India, New Delhi, 2016, pp. 575–582. doi:10.1007/978-81-322-2728-1_54.
URL http://dx.doi.org/10.1007/978-81-322-2728-1_54
- [9] R. C. Gonzalez, R. E. Woods, Digital Image Processing (3rd Edition), Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006.
- [10] L. Armesto, J. Tornero, A. Herraez, J. Asensio, Inspection system based on artificial vision for paint defects detection on cars bodies, in: IEEE International Conference on Robotics and Automation, ICRA 2011, Shanghai, China, 9-13 May 2011, 2011. doi:10.1109/ICRA.2011.5980570.
URL <http://dx.doi.org/10.1109/ICRA.2011.5980570>
- [11] S. Werling, M. Mai, M. Heizmann, J. Beyerer, Inspection of specular and partially specular surfaces, Metrology and Measurement Systems 16 (3) (2009) 415–431.
- [12] Micro-Epsilon, Fully automatic surface inspection of painted car bodies (2015).
URL <http://www.micro-epsilon.com/measurement-systems/Paint-Inspection/karosserie/index.html>
- [13] ISRA, Paintscan: Paint inspection with in-line deflectometry (2015).
URL http://www.isravision.com/media/wm/public/prospekte2014/paintscan3d/Flyer_PAINTSCAN3D_en.pdf
- [14] J. Tornero, L. Armesto, M. C. Mora, N. Montés, A. Herráez, J. Asensio, Detección de defectos en carrocerías de vehículos basado en visión

- artificial: Diseño e implantación, *Revista Iberoamericana de Automática e Informática Industrial RIAI* 9 (1) (2012) 93–104. doi:<http://dx.doi.org/10.1016/j.riai.2011.11.010>.
URL <http://www.sciencedirect.com/science/article/pii/S1697791211000793>
- [15] M. Imanishi, K. Yoshida, T. Asaeda, Y. Suzuki, S. Chida, M. Watanabe, Surface defect inspection apparatus, uS Patent 5,726,705 (Mar 1998).
URL <https://www.google.ch/patents/US5726705>
- [16] K. Alders, M. Lehe, G. Wan, Method for the automatic recognition of surface defects in body shells and device for carrying out said method, uS Patent 6,320,654 (Nov 2001).
URL <https://www.google.ch/patents/US6320654>
- [17] D. Clarke, R. Reynolds, T. Pryor, Panel surface flaw inspection, uS Patent 4,629,319 (Dec 1986).
URL <https://www.google.com/patents/US4629319>
- [18] B. Denkena, H. Ahlers, F. Berg, T. Wolf, H. Tonshoff, Fast inspection of larger sized curved surfaces by stripe projection, *CIRP Annals - Manufacturing Technology* 51 (1) (2002) 499 – 502. doi:[http://dx.doi.org/10.1016/S0007-8506\(07\)61570-4](http://dx.doi.org/10.1016/S0007-8506(07)61570-4).
- [19] D. Fortun, P. Bouthemy, C. Kervrann, Optical flow modeling and computation: A survey, *Computer Vision and Image Understanding* 134 (2015) 1 – 21, *image Understanding for Real-world Distributed Video Networks*. doi:<http://dx.doi.org/10.1016/j.cviu.2015.02.008>.
URL <http://www.sciencedirect.com/science/article/pii/S1077314215000429>
- [20] S. Kammel, F. Puente León, Deflectometric measurement of specular surfaces, *IEEE Transactions on Instrumentation and Measurement* 57 (4) (2008) 763–769. doi:[10.1109/IMTC.2005.1604173](https://doi.org/10.1109/IMTC.2005.1604173).

- [21] J. Santolaria, J. Velázquez, D. Samper, J. Aguilar, I. Escursell, Sistema de inspección de defectos opel españa (2016).
URL <http://www.opel.es/acerca-de-opel/noticias-opel/2016/septiembre/i3a-y-gm-sistema-deteccion-defectos-pintura-coches.html>
- [22] A. A. Goshtasby, 2DD and 3-D Image Registration: For Medical, Remote Sensing, and Industrial Applications, Wiley-Interscience, 2005.
- [23] S.-s. Chen, Stochastic and neural methods in signal processing, image processing, and computer vision, International Society for Optical Engineering, vol. 1766 of Proceedings of SPIE, (San Diego, CA), July 1992, 1992.
- [24] A. Bruhn, J. Weickert, C. Schnörr, Lucas/kanade meets horn/schunck: Combining local and global optic flow methods, International Journal of Computer Vision 61 (3) (2005) 211–231. doi:10.1023/B:VISI.0000045324.43199.43.
URL <http://dx.doi.org/10.1023/B:VISI.0000045324.43199.43>
- [25] B. D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, in: Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2, IJCAI'81, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1981, pp. 674–679.
URL <http://dl.acm.org/citation.cfm?id=1623264.1623280>
- [26] N. Dalal, B. Triggs, C. Schmid, Human detection using oriented histograms of flow and appearance, in: Proceedings of the 9th European Conference on Computer Vision - Volume Part II, ECCV'06, Springer-Verlag, 2006, pp. 428–441. doi:10.1007/11744047_33.
- [27] T. Kiuchi, K. Ikeuchi, Roughness and shape of specular lobe surfaces using photometric sampling method, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 1993, pp. 765–766. doi:10.1109/CVPR.1993.341164.

- [28] Zeiss, Abis - automatic body inspection system (2015).
URL <https://www.itwm.fraunhofer.de/en/departments/image-processing/surface-inspection/abis-automatic-body-inspection-system.html>
- [29] A. Yogeswaran, P. Payeur, 3D Surface Analysis for Automated Detection of Deformations on Automotive Body Panels, INTECH Open Access Publisher, 2012.
URL <https://books.google.es/books?id=k4Z9oAEACAAJ>
- [30] V. Borsu, A. Yogeswaran, P. Payeur, Automated surface deformations detection and marking on automotive body panels, in: 2010 IEEE International Conference on Automation Science and Engineering, 2010, pp. 551–556. doi:10.1109/COASE.2010.5584643.
- [31] G. Bergmann, Á. Horváth, I. Ráth, D. Varró, A benchmark evaluation of incremental pattern matching in graph transformation, in: H. Ehrig, R. Heckel, G. Rozenberg, G. Taentzer (Eds.), Proc. 4th International Conference on Graph Transformations, ICGT 2008, Vol. 5214 of Lecture Notes in Computer Science, Springer, Springer, 2008, pp. 396–410, acceptance rate: 40%.
- [32] A. Buades, B. Coll, J.-M. Morel, A non-local algorithm for image denoising, in: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 - Volume 02, CVPR '05, IEEE Computer Society, Washington, DC, USA, 2005, pp. 60–65. doi:10.1109/CVPR.2005.38.
URL <http://dx.doi.org/10.1109/CVPR.2005.38>
- [33] Nvidia, What is gpu-accelerated computing? (2007).
URL <http://www.nvidia.com/object/what-is-gpu-computing.html>