

Visualization of Industrial Engineering Data in Augmented Reality

Bruno, F.* , Caruso, F.* , De Napoli, L.* and Muzzupappa, M.*

* Università della Calabria, Dipartimento di Meccanica, 87030 Arcavacata di Rende (CS), Italy.
E-mail: {f.bruno, francesco.caruso, ldenapoli, muzzupappa}@unical.it

Received 20 April 2005
Revised 26 November 2005

Abstract: This paper presents an innovative application of Augmented Reality (AR) techniques in the field of industrial engineering in which the user explores data from numerical simulations or the results of measurements and experiments, superimposed to the real object that they refer to. The user observes the object through a tablet PC, used as a video see-through handheld display. Data are visualized superimposed to the real object that represents a spatial reference relative to which the user can refer to, so the exploration is more natural compared to a traditional visualization software. Moreover, we have developed a new framework, called VTK4AR, that provides a set of useful software classes for the rapid development of AR applications for scientific visualization. VTK4AR is built on top of VTK (an open source API for scientific visualization), so it will be possible to employ a wide range of visualization techniques in many application fields, and moreover, it is possible to interactively manipulate data-sets in order to achieve a more effective way of visualization.

Keywords: Augmented reality, Scientific visualization, Interaction techniques.

1. Introduction

Augmented reality (AR) supplements reality by allowing the user to perceive the real world with virtual objects superimposed upon it or merged with it. Virtual objects convey information that the user cannot directly detect with his/her own senses. AR applications have been implemented in many application domains such as assembly, maintenance and repair of complex equipment (D. Curtis et al., 1998; Navab et al., 1999), cultural heritage (Dähne et al., 2002) and entertainment (Thomas et al., 2000). A comprehensive overview of AR could be found in Azuma (1997) and Azuma et al. (2001).

AR techniques have been employed also in some scientific visualization tasks like in some medical applications for the visualization of CT (Computed Tomography), MRI (Magnetic Resonance Imaging) or ultrasound scans. In surgery planning, the required view of the internal anatomy is projected over the target surface to help the surgical procedure (Grimson et al., 1998). In ultrasound imaging it is e.g., possible to display the volumetric rendered image of the foetus overlaid on the abdomen of the pregnant woman. The image appears as if it were inside the abdomen and is correctly rendered as the user moves (Andrei et al., 1994; Sauer et al., 2001).

Other applications of scientific visualization in AR have been developed upon Studierstube, a framework for collaborative AR (Szalavari et al., 1998; Schmalstieg et al., 2002; Fuhrmann et al., 1998; Barakonyi et al., 2003). These applications allow users to display and examine volumetric data in a collaborative AR environment, and to highlight or explore slices of the volume by manipulating

an optical marker as a cutting plane interaction device. In these applications AR is a novel interface for scientific visualization for improving collaborations among the users, but it has not been employed to enrich real objects with useful information related to the objects itself.

From the papers mentioned above, it can be inferred that AR visualization improves comprehension and interaction of the visualized data. However, none of the authors above employs this technique in the industrial engineering field, in which more and more effective ways to analyze data are required. From this perspective, in fact, the AR visualization improves the collaboration among designers and technicians, since their analyses tasks are often correlated. Furthermore, it improves the handling and availability of the data itself and also the need to explain complex data-sets to non-expert people is simplified.

With this in mind, we have developed a methodology that provides the tools and defines the steps needed to obtain the visualization in AR of industrial engineering data.

The methodology employs different engineering technologies, an AR system based on a tablet PC as a displaying device and a new all-in-one framework, called VTK4AR, that allows developers to easily and rapidly develop AR applications for scientific visualization.

In accord with what is above mentioned, this paper presents two main innovations:

1. For the first time AR techniques are employed in data visualization in an industrial design context.
2. A new AR software framework for the rapid development of AR applications for Scientific visualization has been developed.

2. Methodology

The introduction of computer aided systems (CAx systems) has made the use of virtual prototypes indispensable, thus radically modifying the design process of industrial products. On the other hand, despite the fact that such systems are nowadays widespread, design activities still strongly depend on physical mock-ups, which are necessary to validate numerical models.

The need to use both physical and virtual prototypes during the design activity has supported the development of technologies able to easily generate physical prototypes through virtual ones and vice versa. In particular, the Reverse Engineering (RE) is nowadays the most common technique used in the engineering fields to generate virtual models from physical models, through a 3D scanner; whereas the Rapid Prototyping (RP) is the technique which allows an easy and completely automated way of generating physical prototypes through virtual models.

The methodology, proposed in the present study, integrates the above-mentioned technologies with AR, for the first time. It stands out because it has two starting points according to the kind of prototype (virtual or physical) designers initially have at their disposal.

The methodology in question requires the presence of both prototypes and, therefore, the first step one has to take is the generation of the missing prototype through the above-mentioned techniques. Secondly, in keeping with project requirements, one has to decide whether to use the virtual model to make simulations or analysis (e.g., structural, fluid-dynamic, thermal, etc.) or the physical model to carry out experiments and measurements (e.g., assessing mistakes made during the forming process, tension and fatigue tests, wind tunnel analyses, etc.). The third step is the post processing of data obtained, both through CAE analysis and experiments and measurements. The last step is the visualization in AR, which is made through an ad hoc software: VTK4AR. This software is intended to create an AR environment which uses a special hardware set-up and video-tracking technology in order to be able to visualize the data-set directly superimposed on the real model. Moreover, VTK4AR requires the contemporaneous use of both the physical and the virtual prototype. In particular, VTK4AR uses virtual models to correctly visualize the data-set, solving any occurring occlusion problems. Furthermore, the methodology uses the physical model to define the spatial reference on which the results of the simulation are visualized.

In Fig. 1 the various steps of the methodology proposed, are shown.

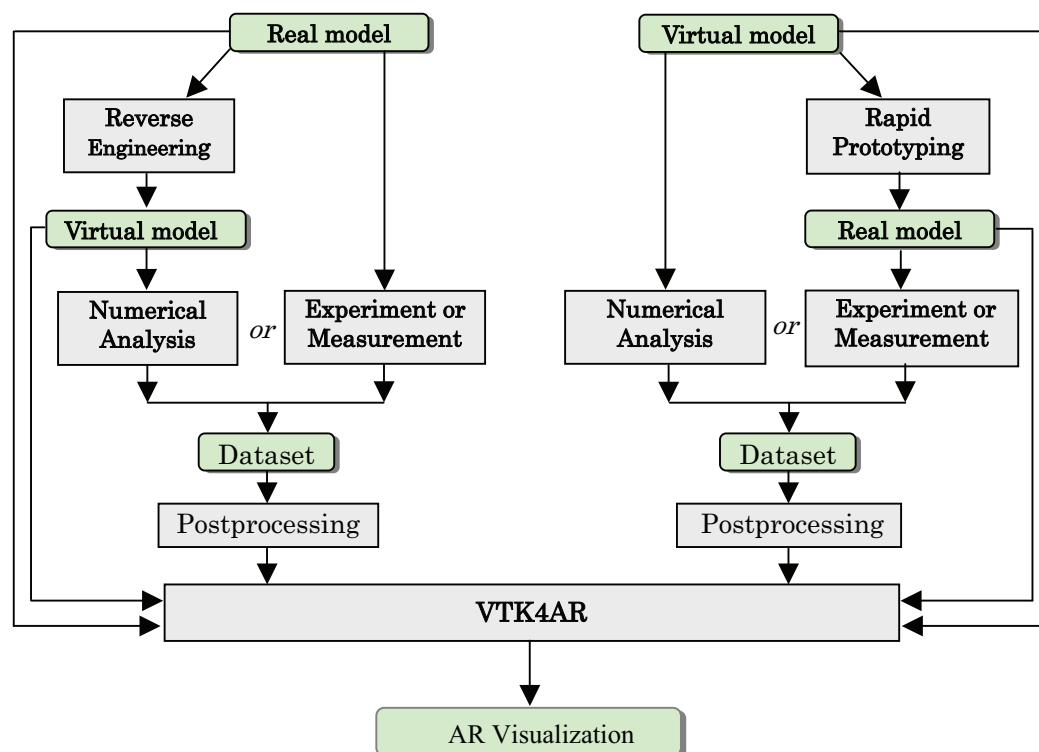


Fig. 1. Methodology Diagram.

3. Hardware Set-Up

An AR system can be implemented using different hardware configurations. One of the most widespread solutions is the optical see-through Head Mounted Display (HMD). Another chance is the video see-through approach: one or two cameras can be employed to transform a normal HMD, a hand-held device (Schmalstieg, 2003) or a flat monitor (Rekimoto, 1997), into a simulated see-through display.

Tablet PC and PDA have been widely employed in AR applications. In (Abawi et al., 2004) the Tablet PC was employed to display detailed information about plant machines in an oil refinery (i.e., the inner parts of a pump), and, in the real in real project (www.contents4u.com), tablets and PDAs are employed in interior design applications to show virtual furniture in a real room.

We have employed a tablet PC as a display device because it offers some advantages compared to other possible configurations. Compared to the HMD, it is a low cost device, it is completely wireless and can be observed by two or three people at the same time, but, on the other hand, it cannot support stereoscopic vision. Compared to a handheld device, it offers a wider display and much better hardware performances, but it weighs more.

As described by Rekimoto (1995), the tablet works like a magnifying glass, that can be used to perceive information about the objects that, otherwise, could not be seen. The tablet can be imagined as an augmenting window which augments the perception of the objects visualized through it. Navigation is then much more natural compared to a standard desktop environment. Furthermore, real objects on which the data are superimposed represent a spatial reference which the user can refer to. But the magnifying glass metaphor in industrial engineering data visualization has never been tested before.

The tablet PC is equipped with a video camera employed both to simulate the see-through functionality and to track the display position with a pattern recognition algorithm.

The hardware used in this study is a tablet PC with: Intel® Pentium® M processor 1.60 GHz,

1024MB PC2700 DDR333 SDRAM, Nvidia GeForce FX Go5200 32Mb graphics card, 12.1" Polysilicon SXGA+ Display, weight 2.0Kg. The webcam employed is a Creative LivePro. It has a VGA CCD sensor and a digital video capture speed of 30 fps.

The video camera has been mounted on the top of the tablet, so it is possible for the user to handle the tablet with one hand, while the other hand holds the pen used to interact with widgets displayed on the tablet monitor, as it can be seen in Fig. 2.

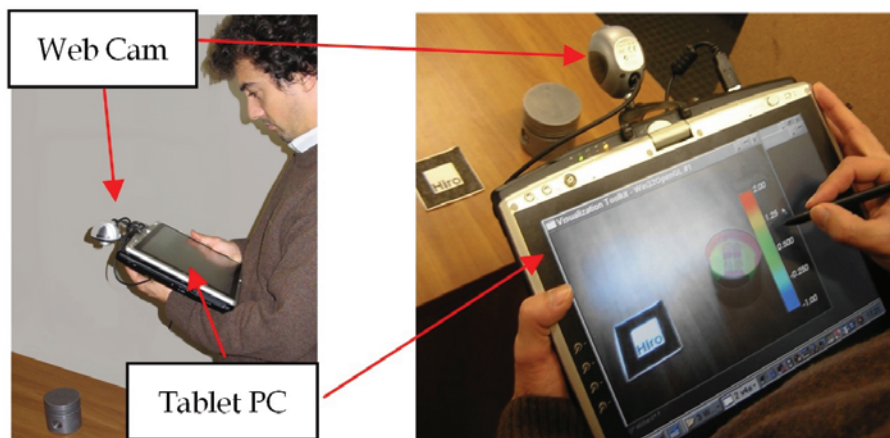


Fig. 2. Hardware set-up of vtk4ar.

The use of the tablet PC as a magnifying glass allows the user to immediately become familiar with the system, even if he/she does not have any experience in AR. The exploration of the dataset is natural because the user only has to move the tablet where he/she would like to display data. In fact the magnifying glass metaphor gives the user the possibility to directly experience the 3D space, and this is the best way to understand the 3D space itself (Hinckley et al., 1994). The real object on which data are superimposed, represents a spatial reference, which the user can refer to when interacting in 3D. In fact, as also reported in an experiment, made by Badler et al. (1986), the presence of a real spatial reference can change the 3D interaction task from a “consciously calculated activity” to a “natural and effortless” action.

4. The VTK4AR Framework

A video-tracking based AR application for scientific visualization needs to perform four fundamental operations: video capture, user position tracking, reading and managing data-set and a correct visualization of the AR scene. A general implementation schema of a video-tracking based AR application is shown in Fig. 3. In VTK4AR the first two tasks are performed using the video-camera handling, and video tracking functionalities of the ARToolKit library suite (www.hitl.washington.edu/artoolkit), considered by now a standard de facto for AR. Instead, as regards the management of data-sets and the visualization of the AR scene, we have used VTK (www.vtk.org), a specific library for scientific visualization.

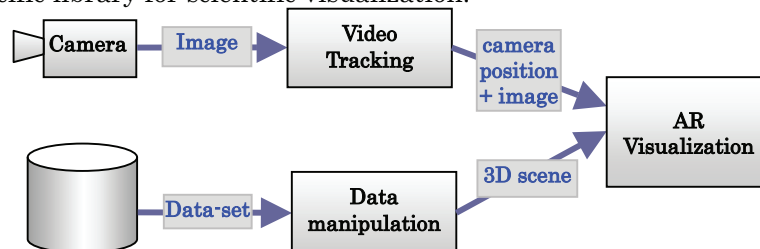


Fig. 3. Implementation schema of a video-tracking based AR application for scientific visualization.

There are many free libraries that perform such operations, but we have employed only open source software both to keep license costs down and for the possibility to publish them achieving a large diffusion of the VTK4AR framework. As a matter of fact, it will soon be released under GNU/GPL license. Both libraries are cross platform, so all the porting problems have been bypassed (actually VTK4AR has been tested on Windows and Linux).

4.1 Implementation

VTK4AR is born to be a framework to visualize all kinds of datasets in AR with the availability of all the functionalities implemented in VTK. VTK4AR is made up by a set of high level software classes to simplify the development of new AR applications. It can be considered as an extension of VTK with the support for AR. In fact, VTK has been extended with some classes, specific for the scientific visualization in AR. Furthermore, a set of classes for an easy and object oriented handling of the video camera and tracking system has been coded.

Compared to the medical applications (Grimson et al., 1998; Andrei et al., 1994; Sauer et al., 2001), in which the virtual data and the real images were mixed by the hardware, VTK4AR is able to completely handle the application, natively supporting the video see-through and the video-tracking functionalities. Also compared to the technology used in the Studierstube applications, VTK4AR represents an improvement. In fact, in the above-mentioned works, there was a total decoupling between the scientific visualization system and the AR user interface (Studierstube). The AR scene was obtained exporting visualization data from the visualization system to Studierstube. As the authors also recognize, this approach has a bottleneck in the data conversion process that does not have to be repeated each time that visualization parameters change. Instead, VTK4AR allows a rapid implementation of all the interaction techniques for data manipulation, without any conversion.

4.1.1 The Occlusion Problem

VTK4AR also manages the occlusion between real and virtual objects. As a matter of fact, one of the problems in AR visualization arises when real objects occlude 3D geometries, thus leading to a misunderstanding of the AR scene. In the case of CFD (Computational Fluid Dynamics) analysis of a cylinder such misunderstanding is shown in Fig. 4(a). If the streamlines are drawn superimposed on the real scene, the result is not realistic, because the streamlines “do not wrap” the cylinder. In Fig. 4(b) the correct occlusion of the cylinder has been calculated. The data-set, employed in the example represented in Fig. 4, is a demo included in the VTK package.

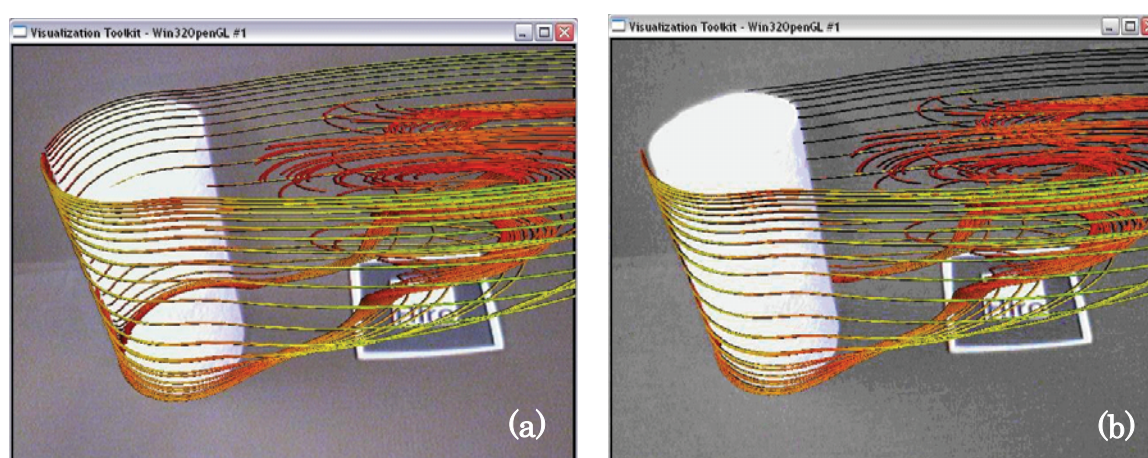


Fig. 4. Streamlines wrapping a real cylinder without (a) and with (b) occlusion taken in account.

In previous works the occlusion problem has been tackled with different methods, but two general approaches can be identified. The first approach, defined as “model-based”, implies that geometries of real objects are known. It means that the model of the real object, correctly positioned in the virtual world, can be used to produce occlusions by drawing it in black. These models have to be placed in the correct position in camera coordinates to produce an image identical to the live video image of the real objects. In other words, when the geometric model is rendered, it appears at the same location and orientation in the computer-generated image as the real object in the live video. In the system presented in (Breen et al., 1996) the video output of the graphics workstation is combined with the output of a video camera using luminance keying.

If the geometric models of the real objects are unknown, the depth-based approach can be employed (Fuhrmann et al., 1999; Berger, 1997; Lepetit and Berger, 2000; Wloka and Anderson, 1995). Depth information has to be written in the Z-Buffer before the rendering of the virtual objects occurs. With this approach, the model of the real world is acquired reconstructing the depth of the real environment from the point of view of the camera.

In our methodology, the geometrical models of the real objects are available, so we have chosen a model based approach. Occlusion handling has been implemented thanks to a proper class, which writes in the Z-Buffer only. The matter will be analyzed in depth later.

4.1.2 The *ARToolkitHandle* Library

In order to have an easy handling of the tracking and video libraries, and to have a full Object Oriented environment, we have developed *ARToolkitHandle*, a C++ API for managing *ARToolkit* C libraries.

The *ARToolkitHandle* library is made up of two main classes:

- *ARVideoHandle* class that provides an easy interface to the video devices;
- *ARVideoTracker* that is an abstract class, implemented in *ARMonoMarkerTracker* and *ARMultiMarkerTracker* classes, for managing mono-marker and multi-marker tracking.

Every *ARVideoTracker* object has associated an *ARVideoHandle* object, from which it acquires images that will be processed for tracking.

4.1.3 New *VTK* Classes for *AR*

The technique to obtain an *AR* scene is quite standard: virtual objects (i.e., the results of CAE analysis) must be rendered above the real scene obtained by the camera. To automate the development of new applications, *VTK4AR* directly provides the developers an environment made up by:

- a viewport to render virtual objects (*vtkRenderer*)
- a viewport to render the real image (*vtkARImageRenderer*)
- a window with two layers to contain the previously mentioned viewports (*vtkARWindow*).

The *vtkRenderer* class is a native class of *VTK* that renders actors (e.g., 3D objects, data-set and images) in a window. Every *vtkRenderer* has its own camera, lights and actors.

The new class *vtkARImageRenderer* (directly derived from the *vtkRenderer* class) renders the camera grabbed images in the window. Every *vtkARImageRenderer* is linked by default to a *ARVideoHandle* object, which manages the camera, to generate the background image. Finally, the new class *vtkARWindow* contains a *vtkARImageRenderer* in the background and a *vtkRenderer* in the foreground.

In order to be able to display the occlusion between real and virtual objects correctly, a new class called *vtkARDepthActor* has been implemented. The main feature of this kind of actor is that it will be rendered in the Z-Buffer only. *VtkARDepthActors* must be rendered before the standard *vtkActor*. In this way, the depth-buffer is filled with the Z values of real objects, before virtual objects are rendered, and, therefore, only the visible parts of the virtual objects will be drawn.

A complete UML class diagram of the *VTK4AR* framework is shown in Fig. 5.

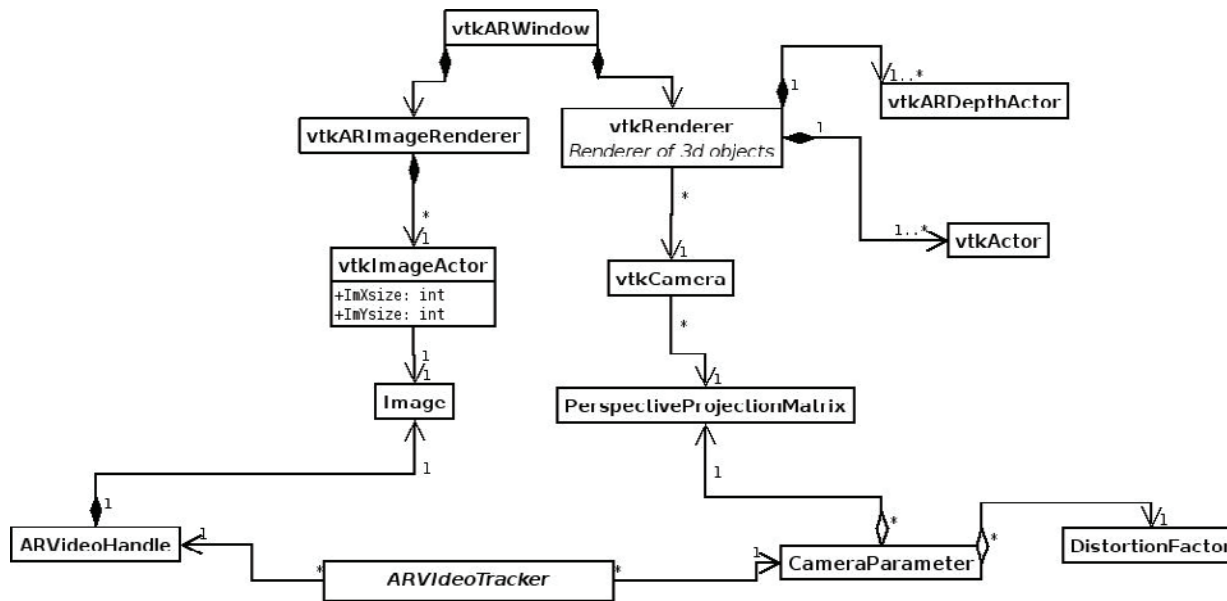


Fig. 5. UML class diagram of the framework.

5. Case Studies

In this section two case studies of our methodology are presented.

The first case (Fig. 6) shows the results of a CFD simulation of the flow past a helmet. The application of our methodology to this case has four steps:

1. the CAD model of the product is obtained by means of reverse engineering techniques, in particular a 3D laser scanner Minolta Vivid 300 (VI-300) has been employed;
2. a CFD analysis is performed on this model using COSMOSFloWorks (www.solidworks.com) and the results have been exported in ASCII format;
3. the results are imported, through a data reader developed ad-hoc, in Paraview (www.paraview.org). It is an open-source postprocessor, built on top of VTK and it has been employed to properly choose visualization parameters;
4. the processed data are visualized in AR with VTK4AR.

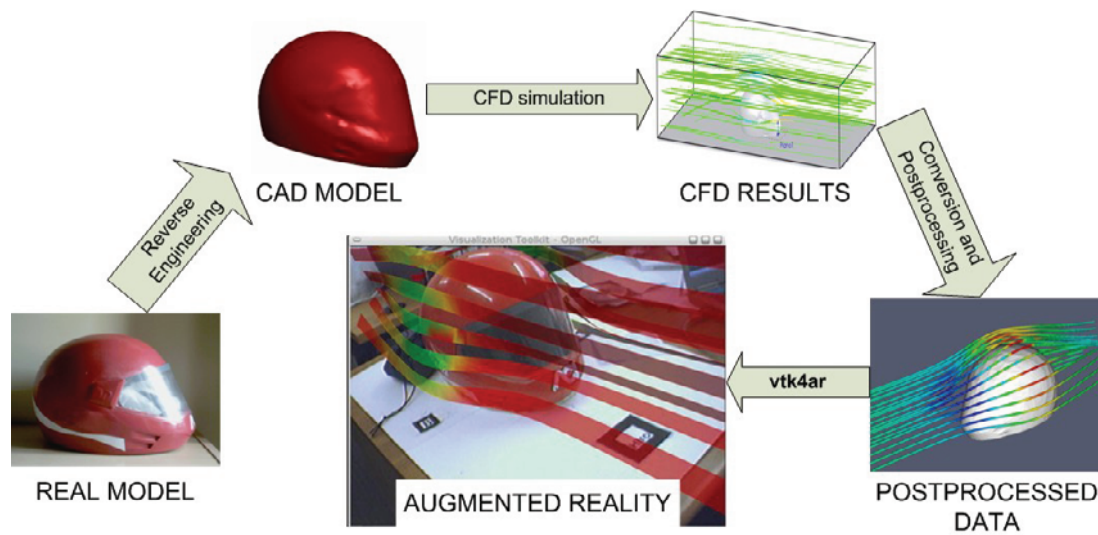


Fig. 6. Schema of the methodology employed in the first test case.

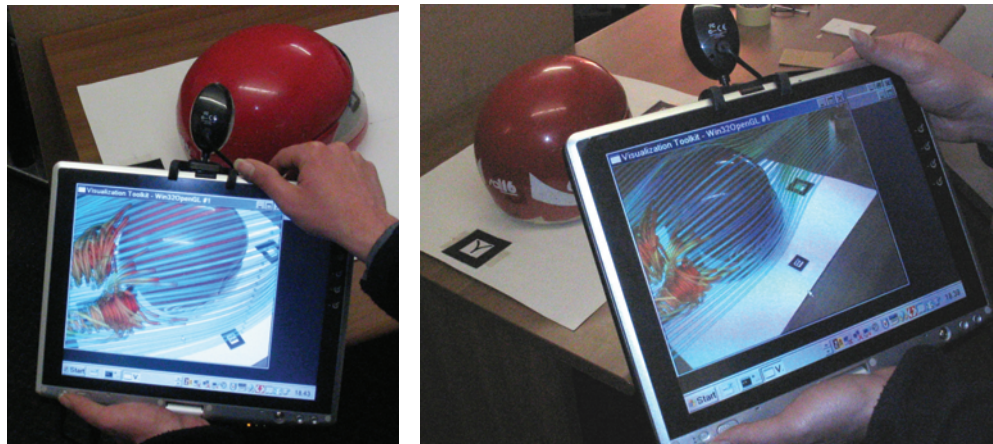


Fig. 7. Virtual streamlines around a real helmet.

The performance of the system, obtained in this test case, are reported in the table below. Two different visualization techniques have been tested: Stream surfaces (central image of Fig. 6) and stream lines (Fig. 7).

Table 1. Performances obtained with two different visualization techniques.

Model	Number of points	Number of Polygons	Frame Rate
Stream-surfaces model	12270	532	~28 fps
Helmet model	1723	3442	
Stream-lines model	29921	58828	~20 fps
Helmet model	1723	3442	

The second test case focuses its attention on the manufacture of an ankle support, which is made starting from what we can roughly call the “patient geometry”. As concerns production technology, Incremental Forming (Filice, 2004) has been utilized; in this kind of application, one of the main drawbacks is represented by the discrepancies between the desired shape and the obtained one. For this reason, a final evaluation of the manufactured product has been carried out at the end of the production step, using AR visualization. The methodology employed in this test case (Fig. 8) has four different steps:

1. the CAD model is generated starting from the patient ankle geometry, acquired by a 3D laser scanner;
2. the prototype of the ankle support is built deforming a metal sheet;
3. the geometry of the prototype is acquired by a 3D laser scanner and the cloud of points, which represents the real product, is compared, in Pro-Engineer environment (www.ptc.com), with the CAD model of the designed surface. This operation generates a map where, for each point, the discrepancy between the designed and the obtained shape is represented by a scalar value (discrepancies map);
4. the discrepancies map is visualized in AR superimposed to the prototype, so the user evaluates the accuracy of the forming process.

The marker needed for the optical tracking has been placed on the physical prototype of the ankle support before the scanning. The texture has been acquired using the camera integrated in the scanner, so the marker appears, drawn on the 3D model surface in the acquisition software. The user can easily determine the position and orientation of the marker in the CAD coordinate system, by

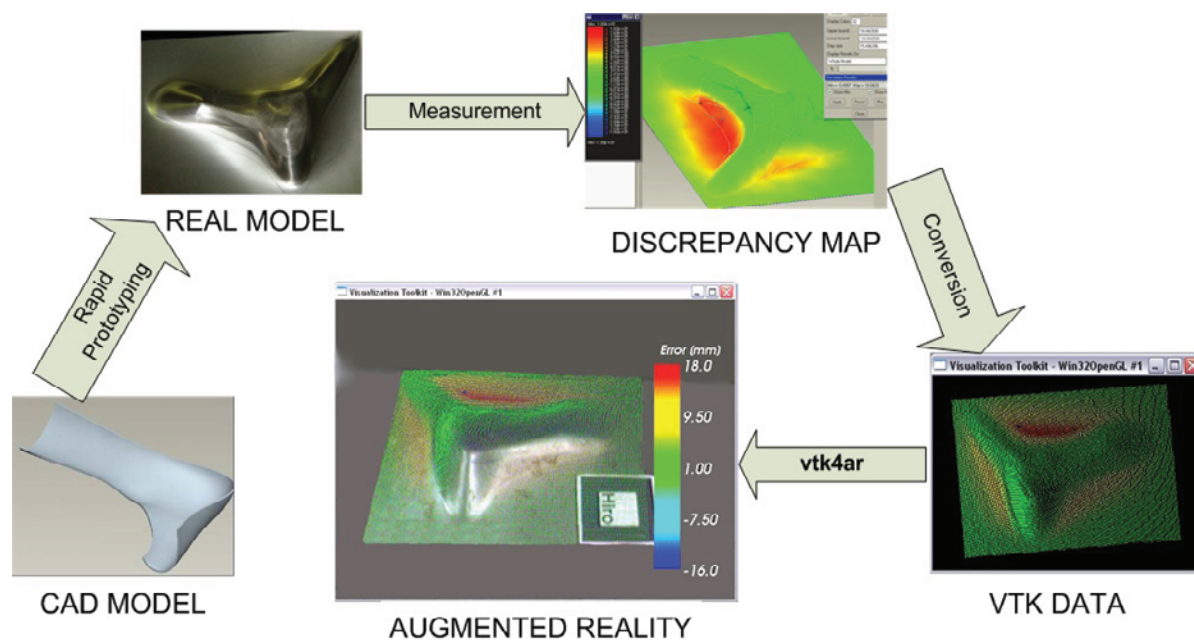


Fig. 8. Schema of the methodology employed in the second test case.

retrieving the coordinates of some points belonging to the marker itself.

The data-set displayed in AR is composed by 15345 points and it is rendered as a point-set, which is a fast rendering technique. The frame-rate obtained is quite high (~29 fps) also because there is no need to consider the occlusion.

Due to the frame grabbing speed of the camera the frame rate of the application cannot exceed 30 fps.

6. Conclusion and Future Developments

Industrial engineering deals with many kinds of data-sets from numerical simulations, measurements or physical experiments. Often these data-sets have to be presented to non-technical people like customers or managers, but this requires that the visualization techniques have to be as easy and clear as possible. This paper presents an innovative methodology that integrates different engineering technologies with AR and scientific visualization, in order to achieve an effective way to present and analyze industrial engineering data. Our methodology allows the data to be displayed directly superimposed on the physical objects, so the user can more easily understand the meaning of the data-sets and their relations with the studied objects. The software framework VTK4AR has been developed to allow the rapid implementation of AR applications, based on this methodology. Moreover, VTK4AR is able to visualize in AR any kind of data supported by VTK, so it could be employed not only in industrial engineering applications, but in every field dealing with scientific visualization.

Test-cases have shown the application of our methodology in two possible fields. The first one visualizes the results of a CFD analysis, employing reverse engineering techniques to obtain the 3D geometry of an existing product. The second one employs a rapid prototyping technology (incremental forming) to obtain a physical prototype starting from a CAD model and visualizes the discrepancy obtained in the forming process. In both the AR environment enriches the meaning of data, since the user perceives without effort the link between real object and the analysis performed on it.

As concerns future developments, we have planned to implement a formal usability test to quantify benefits of this methodology, which must be weighed against the cost of preparation time. In

fact, visualizing data in VTK4AR requires data to be imported in VTK and some alignment of real and digital objects to be carried out as previously described. This task requires an effort that depends on the data complexity, on the scene extension in which data are presented and on the number and the shape of physical objects that can occlude data.

Even if in the industrial practice it is very important to weigh benefits against costs, in some visualization tasks the use of AR, not only improves the human-computer interaction, but it could represent a new useful tool.

We have identified two possible future applications:

1. explain physical phenomena to students from scientific and technical faculties
2. compare numerical and experimental data

The latter is the case of some experimental analyses in which the results could not be expressed with numerical values. For example, in aerodynamic studies in the wind tunnel, techniques like smoke injection, placement of tufts or streamers into the flow (Yang, 1989) and pressure-sensitive paint (Mébarki et al., 2003), (Engler et al., 2005) are used to obtain essential information to be used in the prediction of aerodynamic performance. This data can be retrieved and represented in a virtual environment (Machacek and Rosgenm, 2001), but by using augmented reality techniques the results of the same experiment, realized both physically and numerically, could be visualized at the same time, superimposed on the subject of the experiment. Comparing the results of a numerical simulation and the corresponding physical experiment in AR, the user can perceive possible discrepancies between them immediately, making the refinement of the simulation parameters easier. Moreover, the employment of AR visualization in physical experiments can be crucial when material defects change the expected behavior, when numerical models are not accurate enough in predicting the real behavior.

References

- Abawi, F, Dörner, R, Haller, M., Zauner, J., Efficient Mixed Reality Application Development, CVMP, (2004-3) , 289-294
- Andrei, S., Chen, D., Tector, C., Brandt, A., Chen, H., Ohbuchi, R., Bajura, M. and Fuchs, H., Case study: Observing a volume rendered fetus within a pregnant patient, Proc. IEEE Visualization, (1994-10), Washington, USA.
- Azuma, R., Bailiot, Y., Behringer, R., Freiner, S., Julier, S. and MacIntyre, B., Recent Advances in Augmented Reality, IEEE Computer Graphics and Applications, 21-6 (2001).
- Azuma, R., A Survey of Augmented Reality, Presence Teleoperators and Virtual Environments, 6-4 (1997).
- Badler, N., Manoochehri, K. and Baraff, D., Multi-Dimensional Input Techniques and Articulated Figure Positioning by Multiple Constraints, Proc. ACM Workshop on Interactive 3D Graphics, (1986), 151-170.
- Barakonyi, I., Fahmy, T., Schmalstieg, D. and Kosina, K., Collaborative Work with Volumetric Data Using Augmented Reality Videoconferencing, Proc. IEEE and ACM ISMAR, (Tokyo), (2003-10).
- Berger, M. O., Resolving occlusion in augmented reality: a contour based approach without 3D reconstruction, Proc. IEEE Conference on Computer Vision and Pattern Recognition, (San Juan, Puerto Rico), (1997-6).
- Breen, D. E., Whitaker, R. T., Rose, E. and Tuceryan, M., Interactive Occlusion and Automatic Object Placement for Augmented Reality, Computer Graphics Forum, 15-3, (1996).
- Curtis, D., Mizell, D., Gruenbaum, P. and Janin, A., Several Devils in the Details: Making an AR Application Work in the Airplane Factory, Proc. IEEE Workshop on Augmented Reality, (San Francisco), (1998-11).
- Dähne, P. and Karigiannis, J. N., Archeoguide: System Architecture of a Mobile Outdoor Augmented Reality System, Proc. IEEE and ACM International Symposium on Augmented Reality, (Darmstadt, Germany), (2002).
- Filice L., Ambrogio G., Muzzupappa M., De Napoli L., Fratini L. and Costantino I., Influence of some relevant process parameters on the dimensional accuracy in incremental forming: a numerical and experimental investigation. Journal of Materials Processing Technology, 153-1 (2004), 501-507.
- Engler, R. H., Fey, U., Henne, U., Klein, Chr. and Sachs, W. E., Quantitative Wind Tunnel Studies Using Pressure- and Temperature Sensitive Paints, Journal of Visualization, 8-3 (2005), 277-284.
- Fuhrmann, A., Hesian, G., Faure, F. and Gervautz, M. Occlusion in collaborative augmented environments, Computers and graphics 23-6 (1999-12), 809-819.
- Fuhrmann, A., Löffelmann, H., Schmalstieg, D. and Gervautz, M., Collaborative Visualization in Augmented Reality, IEEE Computer Graphics & Applications, 18-4 (1998), 54-59.
- Grimson, W., Ettinger, G., Kapur, T., Leventon, M., Wells, W. and Kikinis, R., Utilizing segmented MRI data in image guided surgery, International Journal of Pattern Recognition and Artificial Intelligence, 11-8 (1998),1367-1397.
- Hinckley, K., Pausch, R., Goble, J. C. and Kassell, N. F. A., Survey of Design Issues in Spatial Input, Proc. ACM Symposium on User Interface Software and Technology, (1994),213-222.
- Lepetit, V. and Berger, M. O., Handling Occlusion in Augmented Reality Systems: A Semi-Automatic Method, Proc. IEEE and ACM International Symposium on Augmented Reality, (Munich, Germany), (2000-10).
- Mébarki Y., Cooper K. R. and Reichert T. M., Automotive Testing Using Pressure-Sensitive Paint, Journal of Visualization, 6-4 (2003), 381-393.
- Machacek, M. and Rosgenm, T., Development of a quantitative flow visualization tool for application in industrial wind tunnels. Proc. International Congress on Instrumentation in Aerospace Simulation Facilities, (Cleveland, OH, USA),

- (2001-8).
- Navab, N., Bascle, M., Appel and Cubillo, E., Scene Augmentation via the Fusion of Industrial Drawings and Uncalibrated Images with a View to Markerless Calibration, Proc. IEEE Workshop on Augmented Reality, (San Francisco), (1999-10).
- Rekimoto, J., International Conference on Artificial Reality and Tele-Existence '95 / Conference on Virtual Reality Software and Technology (ICAT/VRST '95), (1995-11)
- Rekimoto, J., NaviCam: A Magnifying Glass Approach to Augmented Reality, Presence: Teleoperators and Virtual Environments, 6-4 (1997-8), 399-412.
- Sauer, F., Khamene, A., Bascle, B., Schimmang, L., Wenzel, F. and Vogt, S. Augmented reality visualization of ultrasound images: System description, calibration and features, Proc. IEEE and ACM International Symposium on Augmented Reality, (New York, USA), (2001-10).
- Schmalstieg, D., Fuhrmann, A., Hesina, G., Szalavari, Zs., Encarnação, L. M., Gervautz, M. and Purgathofer, W. The Studierstube Augmented Reality Project PRESENCE - Teleoperators and Virtual Environments, 11-1 (2002-2), 32-54.
- Szalavari, Zs., Schmalstieg, D., Fuhrmann, A. and Gervautz, M., Studierstube - An Environment for Collaboration in Augmented Reality, Virtual Reality - Systems, Development and Applications, 3-1 (1998), 37-49.
- Thomas, B., Close, B., Donoghue, J., Squires, J., Bondi, P. D., Morris, M. and Pieraski, W., ARQuake: An Outdoor/Indoor Augmented Reality First Person Application, Proc. International Symposium on Wearable Computers, (2000).
- Wagner, D. and Schmalstieg, D., First Steps Towards Handheld Augmented Reality, Proc. 7th International Conference on Wearable Computers, (White Plains, NY, USA), (2003-10).
- Wloka, M. M. and Anderson, B. G., Resolving occlusion in augmented reality, Proc. Symposium on Interactive 3D graphics, (Monterey, California), (1995).
- Yang, W. J., Handbook of Flow Visualization, (1989), Hemisphere Publishing, New York.

Author Profile



Fabio Bruno: He is an Assistant Professor at the Department of Mechanical Engineering of the University of Calabria, Italy. He received a Master's degree in Industrial Engineering in 2001. In 2005, he received a PhD degree in Mechanical Engineering at University of Calabria, Italy. The research activity mainly concerns the application of Virtual and Augmented Reality (VR & AR) to the design process of the industrial products.



Francesco Caruso: He is a PhD Student in Mechanical Engineering at University of Calabria, Italy. He received a Master's degree in Mechanical Engineering in 2004. His research focus on Computer Graphics and OpenGL programming especially in scientific visualization tasks, and Augmented Reality.



Luigi De Napoli: He is an Assistant Professor at the Department of Mechanical Engineering of the University of Calabria, Italy. He received a Master's degree in Industrial Engineering in 1993. From 1996 to 2000 he worked at Parametric Technology Italia as Application Engineer. He received a PhD degree in Design and Methods in Industrial Engineering in 2003 at the University of Bologna, Italy. His research activity mainly concerns the application of Reverse Engineering in Mechanical Engineering.



Maurizio Muzzupappa: He is an Associate Professor of Computer Aided Design at the Department of Mechanical Engineering of the Faculty of Engineering of the University of Calabria, Italy. From 1989 to 1992, he frequented his Ph.D. at the Department of Mechanical Engineering of the University of Pisa, Italy. His current research activities include concurrent engineering (specific topics are the computer support of cooperating virtual engineering teams for design review), collaborative design, virtual and augmented reality and reverse engineering.