A Scalable PC-Cluster Architecture for Highly Polygonal Augmented Reality Applications

Dipl.-Inform. Carsten Matysczok*
Heinz Nixdorf Institute
University of Paderborn
Paderborn, Germany

Dipl.-Ing. Michael Grafe#
Heinz Nixdorf Institute
University of Paderborn
Paderborn, Germany

Andrew Wojdala+
ORAD
Hi-Tec Systems Ltd.
Szczecin, Poland

The growing number of Augmented Reality (AR) applications demands high complexity of graphics elements. An example is AR in automotive industry, where there is a need to use models of several million polygons. Even the most powerful today's graphics boards, based on *Ati's Radeon X800* or *nVidia's GeForce 6800* cannot run these models at high frame rates and fall short of the frame rate of video, with which the graphics is mixed.

Therefore we have developed a scalable PC-cluster architecture, which overcomes graphics performance limitations and proposes an efficient method of combining video with graphics, which does not affect the graphics performance of the cluster. Figure 1 shows the architecture of our AR PC-cluster system. The stereoscopic live video is shot by two cameras attached to an HMD and producing two interlaced video signals. These signals are captured by two dedicated nodes equipped with frame grabbers. Each node is processing the captured video in real-time to retrieve the tracking data (for the marker based tracking) and sends them by Ethernet to the rendering nodes.

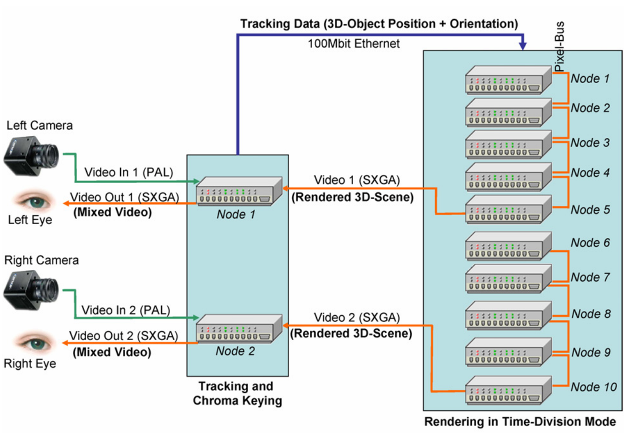

Figure 1. Architecture of the AR PC-cluster system.

The rendering in the perspective determined by the tracking data is performed on a cluster of *DVGs*. DVG is a PC built of COTS (*commercial of-the-shelf*) components expanded with a custom, programmable combiner board. The graphics board (*nVidia GeForce4 Ti 4600*) of each PC is connected to its combiner. Combiners form a daisy-chain using dedicated pixel bus. The last combiner in the chain produces SXGA output. The chain can be configured in different modes; the most efficient is time-division mode, which scales the performance linearly with the number of rendering nodes. For compositing graphics with live video coming from the cameras, a technique similar to chroma keying is used. Chroma keying as a method to combine graphics with live video is well established in certain virtual reality applications, especially in virtual studio. In order to use this technique, capturing nodes are also equipped with combiners. The last node in each rendering chain is connected to the combiner of its capturing node. The combiner was programmed to generate alpha based on keying (background) color. Alpha is then used for blending incoming signal with full-screen polygon mapped with the captured video. The blended image as a SXGA signal is the output and is presented to the user using HMD or shutter glasses in combination with a monitor or a power wall. Figure 2 shows two AR scenes: a wireframe 3D model of a car overlaid on a real car (15 mill. polygons, 1024x768 resolution, 24 Hz frame rate) and a solid 3D model of a car (29 mill. polygons, 1024x768 resolution, 18 Hz frame rate); the markers are used for the optical tracking.

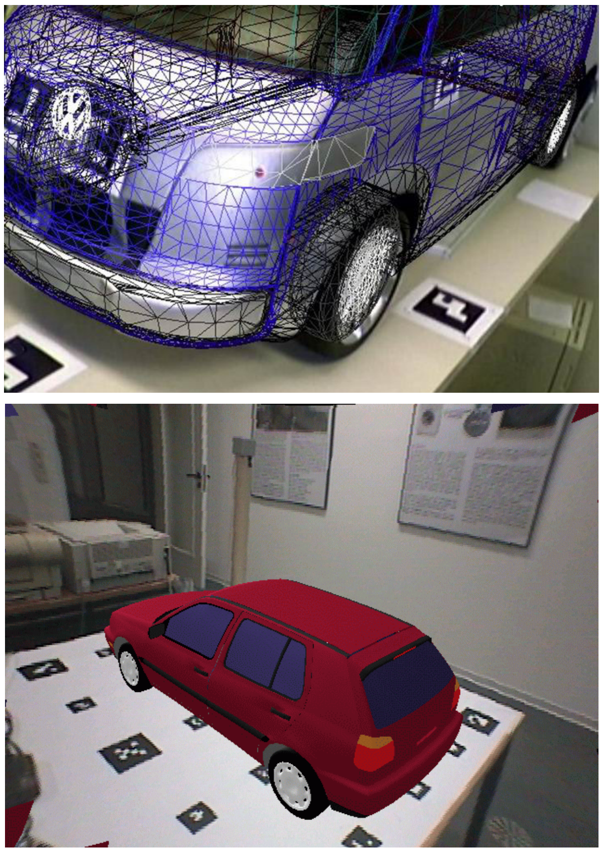

Figure 2. AR scenes: superimposing of reality by highly polygonal 3D models (data coming directly from *Dassault's CATIA V5*).

* email: carsten.matysczok@hni.upb.de
# email: michael.grafe@hni.upb.de
+ email: andrew@orad.pl