Augmented Reality Text Book

Ikraam Elmahguibi, Lujaina Kurwat, Nabil draweel
Computer Engineering Department
Tripoli University
Tripoli, Libya

ABSTRACT: *The idea of this project is to develop a system that interacts with students in the best possible way by bringing class and lab unmanageable physical objects to the tiny personal laptop or a lab PC, this has been implemented using an Augmented Reality Book (ARB). This book will offer students a 3D presentations and interactive experience and will help them take control of their learning at their own pace.*

The ARB will also drive the potential to simulate and motivate students to explore class materials and create a new learning environment suitable for various learning styles; it can help teach subjects where students can't feasibly gain real life experience.

Keywords: Augmented Reality, Education

Received: 1 February 2014, Revised 8 March 2014, Accepted 15 March 2014

1. Introduction

Some scientific experiments such as in physics or chemistry, and real life phenomenon such as weather changes, can be either difficult or expensive to implement. Such Experiments would require significant financial and physical resources, especially for large scale experimentation and audience. This problem even escalates if such experiments need to be explained several times in school labs, or in a repeated self-learning. An innovative and less expensive solution need to be implemented to allow for large scale, close to real life implementation of such experiments and phenomenon. The wider area of e_learning and e_education can play a vital role in bringing these experiments to near real life implementation through intelligent means using an IT based solution.

1.1 Augmented Reality in Education

Information technology is changing education in a number of ways, augmented reality appeals to education approach in that students will learn the most when they take control of their own learning and interact with the augmented reality environments. With Augmented Reality students can manipulate virtual objects or representations of real objects that would otherwise be impossible to interact with. AR is an environment that includes both virtual reality and real-world elements. This environment would include elements from both the real world and a virtual world at the same time, and would be interactive in real time. The virtual objects could be either manipulated or stationary, and can be defined as a combination of real world and virtual elements (computer generated information) which are interactive in real time; it also can be defined as a form of enhancement (it improves or augment what is already there).

2. Related Work

Recently, there has been a proliferation of researches into the use of AR technology in education, the early work of exploring potential of AR to support learning is based on the work of Billing Hurst when he first created the AR popup book, called Magic Book. This later inspired many researchers in various domains. In science learning.

Karawalla et al. developed an AR application to explore the potential of AR to support spatial learning in the study of relationship between earth, sun, moon and the time. The authors performed a comparative study between AR based approaches and the role play activities. The results showed that AR positively assists students in understanding the concept.

Chen investigated how the students interact with AR system and evaluate the perception of students on learning about Amino Acid using AR system called Protein Magic Book (PMB). The results showed that students liked learning using AR since it was portable, provided a clear image of Amino Acid as well as they could observe the model in detail. However, some students found that the AR marker is not easy to use since it developed based on marker recognition, when the student flip the marker, the model disappear.

Physics Playground was developed to facilitate the students in mechanics learning. The system allowed students to actively build their own experiments and study them in 3D. It provides simulation of the experiment.

Live Solar System, was developed and aimed to help the students of form 3 in learning chapter on Star and Galaxies. The system allowed students to explore the solar system. The research focus was on the usability of the system which concentrates on the ease of use, learnability and effectiveness by using pre-test and post-test. The result shown that the system is easy to use and also helps students in understanding the subject of study.

For young children, few works had been done to investigate the potential of AR in supporting learning.

3. System Components

The ARB system consists of hardware components and software components as depicted in Figure 1. For out ARB to work, the end user must have a computer with a webcam, the coded book and our software installed in the computer. In the following we list the system components with further explanation

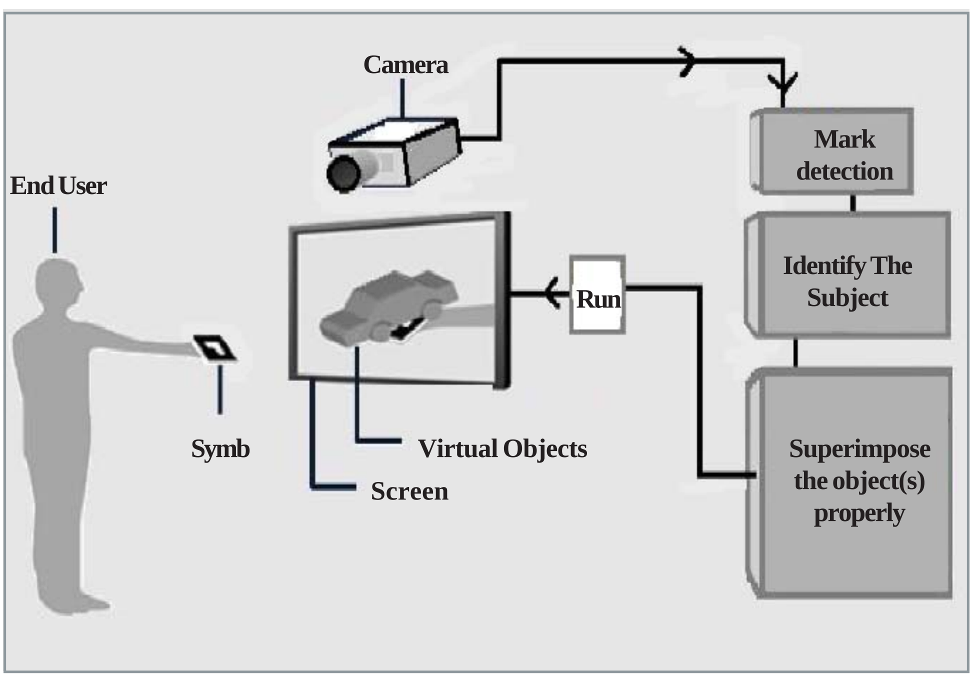

Figure 1. ARB architecture

3.1 Hardware Component (Computer specifications)

- CPU – Intel Pentium 4 2.4GHZ or above or AMD equivalent
- RAM – 1GB memory
- A webcam
- Graphics card – NVidia GeForce FX series or above/ATI Radeon X series or Above /AMD G series chipset or above/ Intel 900G series or above.
- PC Operating System: Windows XP SP2 or above, Windows Vista, Windows 7

4. The Coded Book

A book in which subjects are associated with markers. The function of these markers is to encode the respective subject in it so that ARB can augment the right virtual object. Our coded book is divided into four subsections (space, human body, kids and marketing) just to show different application areas of our software, and the enhancement provided by ARB in different education subjects. We expect that our coded book will be a tool that opens new doors for teachers, and will represent major element in pedagogical development to support teaching ICT native kids.

5. Markers

We can define them as printable patterns that provide AR systems with an easy to recognize reference points, Markers are

Figure 2. Cover page and CD of our coded book

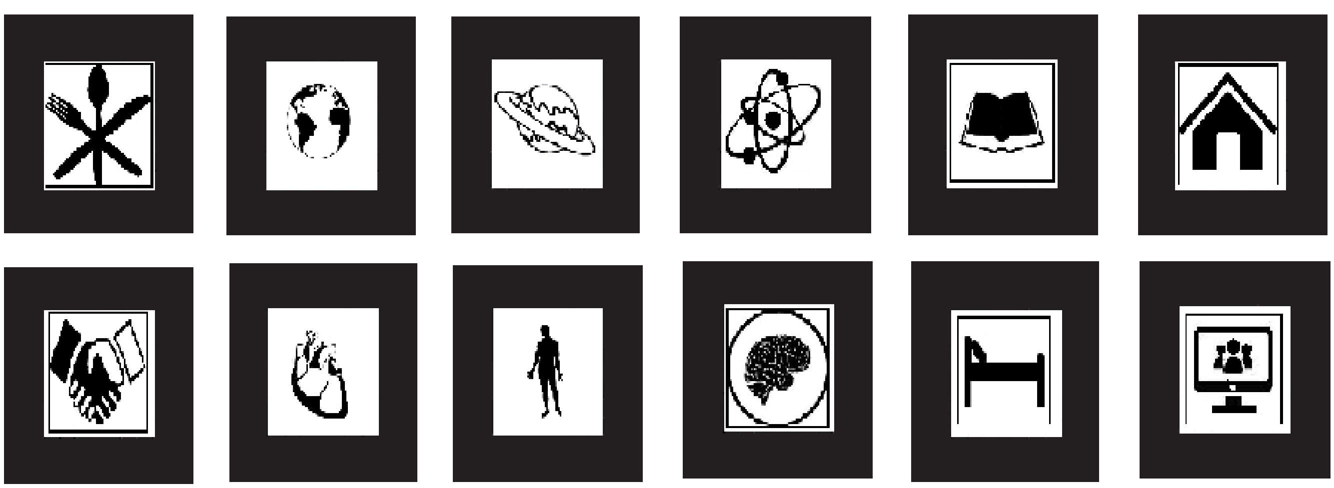

Figure 3. Example of different markers

physical objects or places where the real and virtual environment are fused together; a marker is what the computer identifies as the place where digital information is to be presented [1].

Markers Specifications

Markers need to be square, with a unique image in the middle and black square outline around the image. Only 50% of center area is used in matching process. Marker pattern used in this project is 16 ×16 dots bitmap.

System Implementation

Software components

This component resembles the brain of our project. In order to maintain the intelligence of our ARB, three steps must be performed:

- Marker detection.
- Identifying the object.
- Finally superimposing this 3D object on the top of the marker.

When our system "*sees*" a marker it can determine the marker id and the projection of the camera view. Accordingly, the system retrieves the associated virtual object, and superimposes that virtual object on the marker with the same projection angle.

Implementation Environment and Tools

To proof our concept we used Adobe Creative Suite development environments. We have developed our project in two versions of application. In the first one we used Adobe Flash Professional, which is featured with a SWF file for each marker. This means that it depends on the user to open the file he/she wants to work with manually, where each file is responsible of recognizing certain marker that is associated with one virtual object. In the other version we have put all the markers in one, so there will be only one SWF file for the user to open. The aim in this application is to view all the objects that are associated with the markers in the camera view simultaneously. For the latter application, we used Adobe Flash Builder

FLAR Toolkit

FLARToolKit is mainly developed by Tomohiko Koyama (aka Saqoosha), and has been used in many high profile web-based AR applications. It is a Software library for building AR applications. It also has support for all of the major flash 3D graphics engines (Papervision3D, Away3D, Sandy, and Alternativa3D).

It is an Action Script port of many developing environments, such as NyARToolKit, a Java/C#/Android port of ARToolKit, the well-known AR tracking library developed by Dr. Hirokazu Kato. At Human Interface Technology Laboratory (HIT Lab) at the University of Washington, ARToolKit is the most popular marker based AR tracking library and has been ported to many different operating systems and programming languages. [5] FLARToolKit helps in implementing the first two steps of the software components: marker detection and subject identification.

Papervision3d

Papervision 3D is a 3D engine used with Flash. With Papervision 3D, three-dimensional models can be created in Flash using Actionscript, or models made in other programs can be imported. This model works as skeleton of our virtual object once a model has been created or imported, it can be controlled with Actionscript, just like normal Movie Clip symbols. This makes it possible to create 3D interfaces in Flash, or to create games and applications that use 3D. [6]. However, it is impossible to render the 3D virtual object until it is loaded into the model using specific file type, which is explained below.

COLLADA File

COLLADA (COLLAborative Design Activity) defines an open standard XML schema for exchanging digital assets among various graphics software applications that might otherwise store their assets in incompatible file formats. COLLADA documents that describe digital assets are XML files, usually identified with a .dae (digital asset exchange) extension [7].

Binarizing the input image (Divide image in regions)

First the image is divided in small regions of 16 ×16 pixels and each region is divided into horizontal and vertical scan lines 5 pixels apart.

Labeling (Detect edges in regions)

Then a derivative of Gaussian is convolved on each scan line to estimate the component of the intensity gradient along the scanline. This Gaussian derivative is used to detect black/white edges in the image. The Gaussian derivative is based on convolving the image with a small, separable, and integer valued filter in horizontal and vertical direction and is therefore relatively inexpensive in terms of computations. Local maxima along the scanlines stronger than a certain threshold (for all three color channels!) are considered to be edgels.

Finding squares (segments in regions)

After the detection of the edgels, grouper algorithm is used to construct line segments in each region. It's an algorithm to find groups of "*inliers*", or points which can be fitted into a line. Which works as following:

1. Randomly choose 2 points from the same region, whose orientations are compatible with the line joining the two points, to hypothesize a line.

2. The number of points supporting each line is counted. To be considered part of a line and point must lie close to it and have a compatible orientation with the line.

3. Steps 1 and 2 are repeated ~25 times before a segment is found.

4. Lines with enough support (at least 4 segments) are considered detected lines. Steps 1 to 3 are repeated until all such lines have been found.

5. Because we know the orientation of the edgels from step 2, we can give the line segments an orientation. In the image above, the constructed segments are displayed as red and green arrows.

Merge segments to lines, in this step the segments are merged to lines. Two segments are merged if:

1. The orientations are compatible.

2. The segments are close to each other.

3. If the pixels along the line between the two segments are on an edge, using the same edgel detection criteria as used in step 2.

First all segments are tested in their own region using this algorithm. When all possible segments are merged in their own region, the same operation is repeated for all segments in the whole image.

Extend lines along edges, in this step the lines are extended along the edges. And because we scan only pixels on scan lines 5 pixels apart, it's inevitably that the merged lines don't fit on the entire length of the edge of a marker. So in this step we extend the detected lines pixel by pixel until we detect a corner:

1. Extend each end of a line and check if each pixel is on an edge, using the same edgel detection criteria as used in step 2. Do this until we hit a pixel that's not an edge.

2. Take a pixel a couple of pixels further. If this pixel is '*white*', we might have a found a corner of a black on white marker.

Matching with patterns (Find markers)

Finally, we try to find all markers in the image. To detect the markers in the image, we try to find chains of 3 to 4 lines. A chain is a list of lines where the end of one of the lines hits the start of another. After finding such chains, we only want to keep chains which form a rectangle counter clock wise. These chains will have black '*inside*' and are probably markers.

After finding all markers, for each marker the 4 intersections of the lines in the chain are considered to be the corners. Even if only 3 lines are detected and/or a corner is occluded, the marker will be correctly detected most of the time. In the image below, all detected markers are displayed as red rectangles. Notice the correctly detected marker at the bottom of the image.

Now we have the coordinates of the detected markers, the next step would be to identify markers and distinguish them from each other using Holography transform in order to use this algorithm in an augmented reality application.

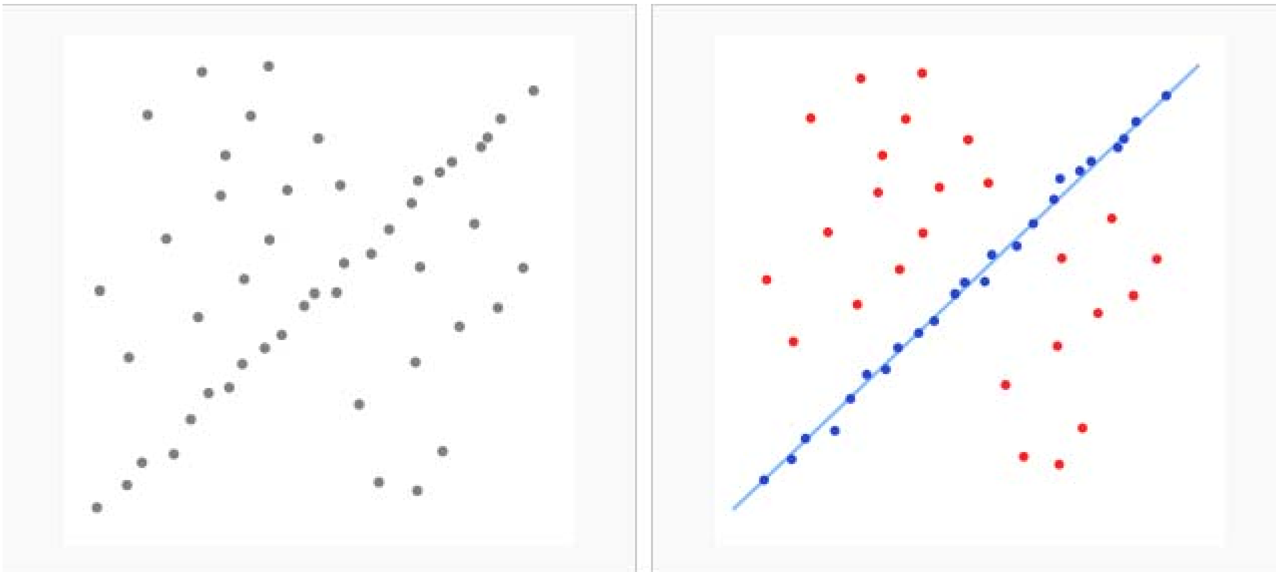

Figure 6. Estimation of scattered patterns

Figure 7. Markers edges recognition

Rendering the 3D objects

FLARToolKit includes support classes which converts FLAR's transform matrix to each 3D engine internal matrix classes.

Playing mp3 sound

Making a sound channel is the basic idea that made this work.

Multiple Markers Loop

The loop part in our software includes all the processing of the data we get from the previous algorithm, the differentiation between the patterns is the key for the code to work properly, and assign each pattern to its model.

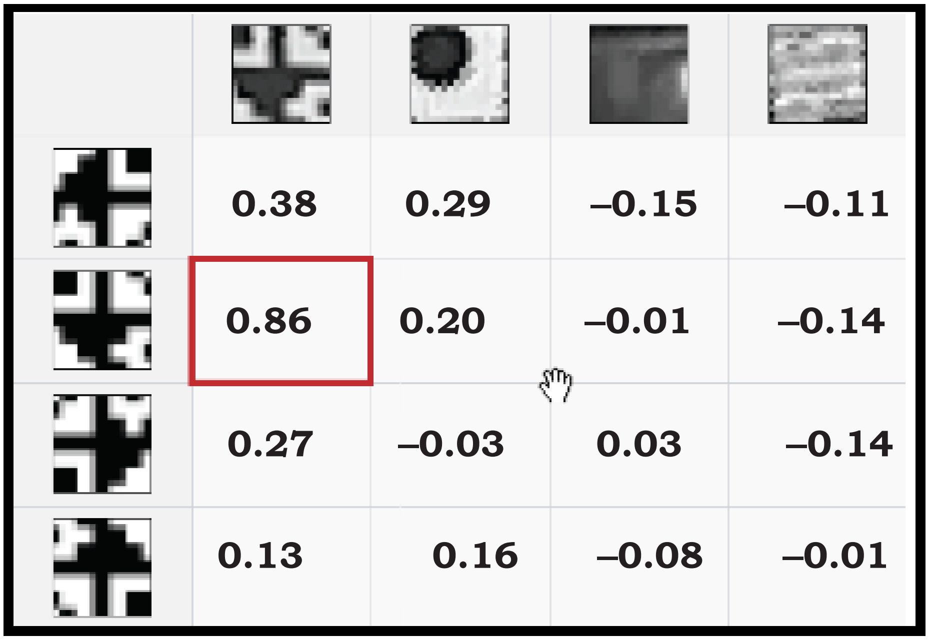

Figure 8. Using thresholds for Homography transform

Figure 9. Rendering the recognized 3D object

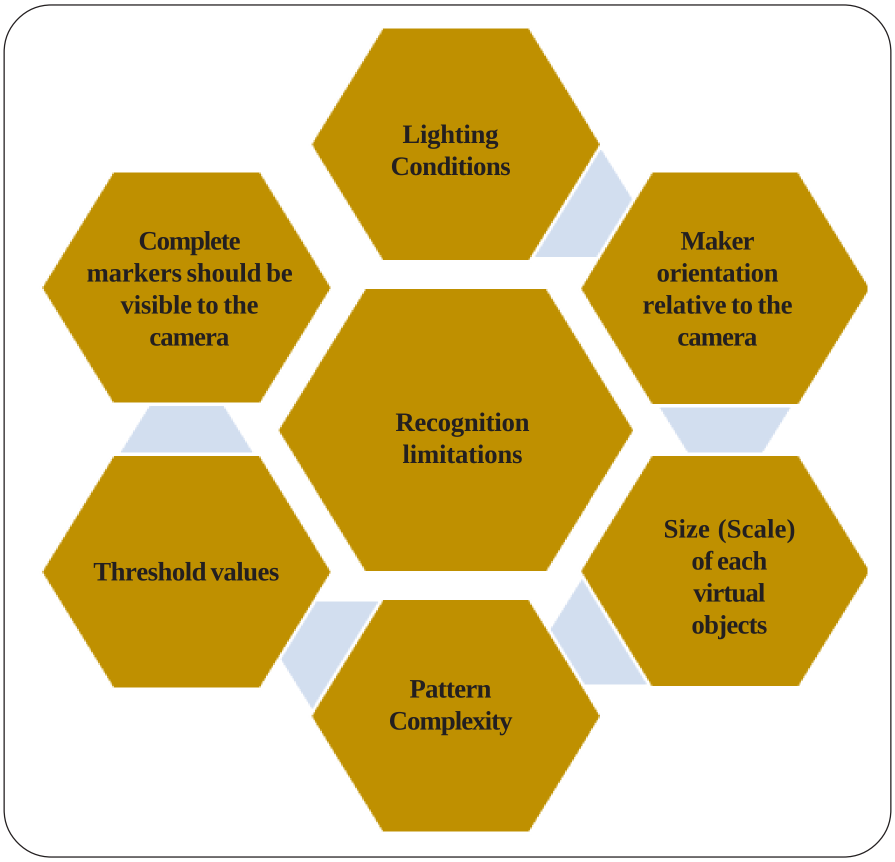

Figure 10. Hexagon radial (limitations)

6. Conclusion

After implementing and testing our project. We have perceived that we met our desired requirements, and we have accomplished the purpose of our Objectives. Which is providing a 3D Augmented Reality experience using our designed (ARB); we managed to test all of the patterns and we could get to an average accuracy of [80%], and an average error of [20%]; so we have managed to create a simplified chart demonstrating the main causes of the error of our system.

7. Future Work

Augmented Reality as new field in both technical development and use, makes it amenable to different possible development paths. In this section we will address two directions of future work.

Different technical enhancements to our work could be applied, some of these are listed below.

1. Enhance the application capability to deal with wide libraries of 3D files such as 3d max, AutoCAD, etc.
2. Re-develop the application to be used on smart phones and tablet PCs, with different operating systems such as Android.
3. Our current development of the application can only accept markers in black and white. This could be enhanced to allow the application to recognize grey-scale images and use the detailed recognition for wider and extensive library of 3D files.

There are several enhancements that can be applied to our project, and on the other hand, our developed application can be used in completely different areas than education such as.

1. In education graphic library of our application could be enriched to include even interactive science experiments, by developing codes that relate to different experiment elements and students can select and connect different elements to construct a complete experiment. This also can be done in math, where animations of basic operations could turn into an interactive graphics, where by students select numbers that can be presented as graphics objects, followed by an operator (+,-, x, /) and then see or derive the results.

2. Away from the Education field, our application can be used as a tool that helps designers and customers to judge how an environment will look after applying specific changes. Codes could represent different construction designs and by pointing the camera to capture a landscape of proposed construction site, the designed block could be induced and the overall look and feel could be accurately judged. This is also possible for decoration, where salesperson could visit a customer's home and by the use of the cameras, and the codes merchandises could be induced on the scene and the customers can have a real life experience on how the end scene will look like.

References

[1] Kipper, G., Rampolla, J. (2013). AUGMENTED REALITY, An Emerging Technologies Guide to AR. Elsevier, Inc.

[2] http://en.wikipedia.org/wiki/Adobe_Creative_Suite

[3] http://en.wikipedia.org/wiki/Adobe_Flash_Builder

[4] http://en.wikipedia.org/wiki/Adobe_Flash_Professional

[5] http://www.artoolworks.com/support/library/FLARToolKit

[6] http://maximized.co.uk/tutorials/papervision1/papervision1a.html

[7] http://en.wikipedia.org/wiki/COLLADA