International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 3 (2011) pp. 480-488

Autostereoscopy and Motion Parallax for Mobile Computer Games Using Commercially Available Hardware

Jens Ogniewski, Ingemar Ragnemalm

Information Coding Group, Linköpings University

581 83 LINKÖPING, Sweden

{jenso|ingis}@isy.liu.se

***Abstract*: In this paper we present a solution for the three dimensional representation of mobile computer games which includes both motion parallax and an autostereoscopic display. The system was built on hardware which is available on the consumer market: an iPhone 3G with a Wazabee 3Dee Shell, which is an autostereoscopic extension for the iPhone. The motion sensor of the phone was used for the implementation of the motion parallax effect as well as for a tilt compensation for the autostereoscopic display. This system was evaluated in a limited user study on mobile 3D displays. Despite some obstacles that needed to be overcome and a few remaining shortcomings of the final system, an overall acceptable 3D experience could be reached. That leads to the conclusion that portable systems for the consumer market which include 3D displays are within reach.**

***Keywords*:** mobile games, autostereoscopy, motion parallax

I. Introduction

By offering a higher immersion, displays allowing stereoscopic vision are expected to play a major role in future entertainment devices. A few such displays are already available on the consumer market. Other important 3D cues like motion parallax can be implemented using readily available soft- and hardware solutions. In this paper we examine how far products available on the mass-consumer market have already come. For that, we chose a typical mobile phone (the iPhone 3G) which provides an optional autostereoscopic enhancement (the Wazabee 3Dee Shell).

Unlike most other papers on 3D displays, the work presented here concentrates on computer games rather than video. Computer games are widely considered to be at least as important as content for 3D enabled devices as three dimensional video. Also, since most computer games already include a detailed 3D description, it is possible to display them directly in stereoscopic vision without the need for alteration. We believe that due to this (and the fact that no three-dimensional video content is available to the customers yet) the first available stereoscopic devices on the market are and will be used mainly for gaming purposes.

The system used for this work was an iPhone 3G which was supplemented with a Wazabee 3dee Shell (a lenticular sheet which can be used to turn an iPhone into an autostereoscopic display). Both are available for purchase on the consumer market. A small animated image which represents a scene of a computer game has been implemented on the hardware, using the accelerometer for the realization of a motion parallax like effect (i.e. tilting the phone changes the perspective of the scene according to the angle the phone was tilted) as well as an autostereoscopic effect using the lenticular sheet. In these cases, care has to be taken to keep up the 3D effect of the lenticular sheet.

The final version of the scene was presented to a few volunteering subjects in different test cases for a small subjective evaluation. The results are presented in this paper as well. Although the hardware used is not the best available, it is sufficient to deliver a solid 3D experience albeit for a few shortcomings.

The remainder of this paper is organized as follows: Section 2 gives an overview of related work. Section 3 describes the system and Section 4 the software solutions that have been developed to overcome shortcomings of the system. Chapter 5 presents the evaluation and Chapter 6 its findings, while Chapter 7 gives some conclusions and suggestions for future work.

II. Related Work

Willner et al.[1] as well as Shi et al. [2] examined portable 3D devices. Both used a modified Nokia N800 Internet Table for their work, and both concentrated on 3D video coding aspects rather than computer games as in this paper. Also, the system used was a prototype system which was never available on the consumer market.

Koike et. al [22] presented a new kind of autostereoscopic display with 60 ray directions which they proposed to apply in mobile gaming applications. However, the technique used in that paper (integral photography) will probably not be available to the consumer for at least a couple of years.

A good introduction to autostereoscopic displays and their advantages and shortcomings as well as some solutions to these is given by Konrad and Halle [3]. Matusik et al. [4] gave a good description on how to build an autostereoscopic display. Meesters et al. [15] examined different artifacts which can occur in autostereoscopic displays, while Wopking [25] developed rules on where to place the objects in relation to the screen to minimize user discomfort.

Jumisko-Pyykkö et al. [5] gave a good overview of user requirements on mobile 3D TV, while Cheng and Nahrstedt [6] examined the properties of autostereoscopic displays. Both results can be applied to mobile computer games using an autostereoscopic display.

Although many papers exist dealing with 3D videos and autostereoscopic displays, and most papers on 3D displays name computer games as a possible application, to our knowledge only one examines their application in computer games [7]. It also concentrates on mobile games, however the study presented there takes a more theoretical approach and does not specify a system.

Mobile computer games have furthermore been evaluated in different publications. Callow et al. [8] give a good overview on the topic, including the 3D graphics system. Nadalutti et al. [9] further emphasized the problems of designing software using the three dimensional graphic accelerator in a mobile device, while Chehimi and Coulton [10] concentrated on the usage of the motion sensor for mobile gaming.

Motion parallax as depth cue is described by Ono [11]. Suenaga et al. and Uehira et al. proved that display systems can be constructed that solely or mostly rely on motion parallax for depth cue [12][13].

Although motion parallax not only enhances the immersion, but could also be part of game-play mechanics, no other article or paper are known to the authors that focuses on computer games using motion parallax. However, two master theses have been done on this topic based on an earlier version of this article: [20], confirms the findings in this article for a desktop solution, and [21], which applies motion parallax in a car driving simulator (which is closely related to racing games and could in fact be described as a serious game application).

Finally, de Vahl [14] showed that it is possible to use conventional games with 3D graphics and render them in stereoscopic vision without the need for their alteration. This was done by capturing the OpenGL function calls and applying the respective function to the two different images needed for stereoscopy. Hence, a lot of content exits already which could be directly used in a stereoscopic vision enabled computer game system for the consumer market, in contrast to stereoscopic video systems.

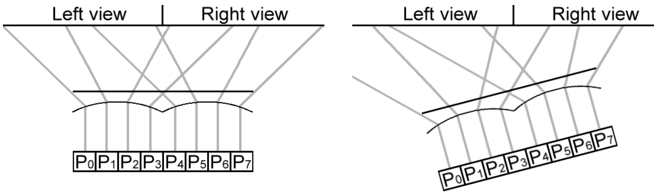

Figure 1. View separation with a lenticular sheet. a) (left) With the phone in its original position, b) (right) after tilting to the left. Note that pixel P2 is assigned to the right view in a), but to the left view in b)

III. System Description

The hardware used for this project was an iPhone 3G. The iPhone platform has two main advantages which made it more suitable for this project than its competitors: it offers a good and easy to use development environment, thus allowing for a quick implementation, and (more importantly) to our knowledge it is the only mobile platform available that actually has a free purchasable enhancement that can be used to display three dimensional images on it, the Wazabee 3Dee Shell.

A. The iPhone 3G

The iPhone 3G is based on a 32bit ARM 11 processor, clocked at 412 MHz. For graphics, it is accompanied by a PowerVR MBX Lite graphics accelerator. Both share 128 MB SDRAM. This configuration and its performance is comparable to many other advanced phones at the time-point of the writing of this article. As in all embedded systems, the computational power is limited and the software designer should be aware of that. Additionally, the MBX Lite only supports OpenGL ES 1.1, which uses a fixed graphics pipeline, instead of Open GL ES 2.0 which offers more flexibility (due to the availability of shaders).

Although the new iPhone 3GS has much more computational power and possibilities (including OpenGL ES 2.0) which would have made this project easier, it was decided to use the slower 3G since work on this project had already started when the 3GS became available.

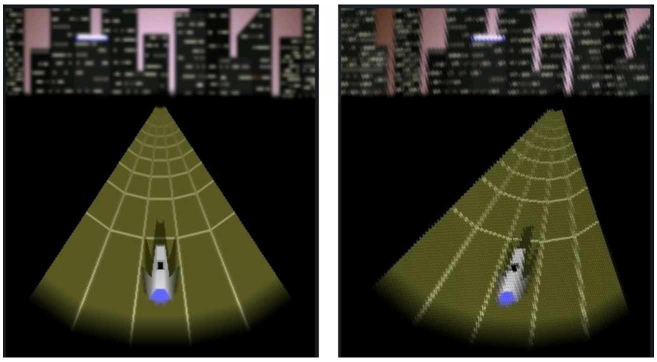

Figure 2. Screenshots from the scene. a) (left) original image, b) (right) after tilting the phone to the left and with distortion as introduced by the stereoscopic mask

B. The Wazabee 3Dee Shell

The Wazabee 3Dee Shell consists mainly of a lenticular sheet and a special iPhone case holding the sheet. The sheet itself can be removed from and reinserted into the case, even while the iPhone is wrapped in it. The sheet does not cover the whole screen of the phone, but leaves a little room at its bottom to allow the implementation of a few buttons on the touchscreen (the touchscreen under the sheet is inaccessible).

The lenticular sheet works like in other autostereoscopic devices. It diffracts the light of the pixel under it in different directions (see also Figure 1a)). Thus, if the phone is hold at the right angle, each eye will receive a different image. Using a special stereo rendering procedure (which will be described in more detail in 4.) it is possible to use this to produce stereoscopic images.

Each lenticule of this sheet is approximately 4 pixel wide and tilted by an angle of approximately 30 degrees to the left to avoid picket fence effect[15], which consists of visible black vertical lines which are induced by an alignment of the borders of the lenticules and the borders of the pixel on the screen.

The 3Dee Shell differs from other autostereoscopic devices in that it might change its position and alignment to the screen between usages since it is removable. A calibration is therefore needed every time the sheet is attached, or otherwise crosstalk [15] will occur. Crosstalk describes the effect if a pixel which should belong to one view influences the other view as well, i.e. that pixels (or some of their color components) can be seen by the wrong eye. Although it might not be possible to remove crosstalk completely, it can be at least minimized through a careful implementation of the autostereoscopic mask, as described in section 4.b).

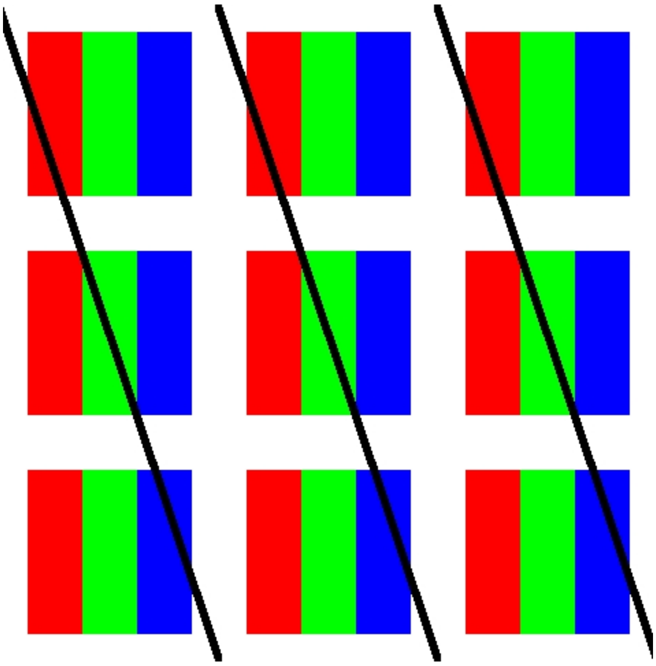

Figure 3. Division of each pixel in its three color channels (RGB), as done physically by the LEDs in the screen. The black lines symbolizes how the tilted lenticules divide the different pixels.

IV. Software Solutions

On the hardware described in Section 3, an animation has been implemented which is shown in Figure 2a). It is composed of a background image (which shows a town), a track and a futuristic vehicle along with its shadow. This might be a scene of e.g. a racing game. This scene has been enhanced to include both motion parallax and autostereoscopic filtering, as described in the following.

Rather than placing objects in front of the screen, the whole scene was placed behind it. This was done to avoid problems with objects that get truncated by the display borders when using the motion parallax effect, which would have reduced the 3D effect and image quality. All objects were placed in the comfortable viewing zone as depicted in [25].

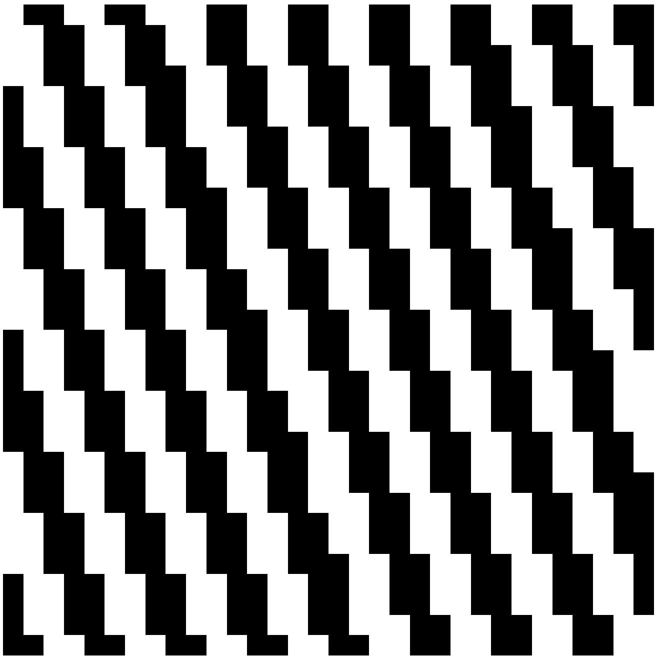

Figure 4. A magnified 32 * 32 pixel large patch of one of the autostereoscopic masks used in this work – in fact the very same which will be called resolution optimized in the evaluation and result sections. The white pixels are allocated to the right view, the black ones to the left view.

A. *Motion Parallax*

As pointed out earlier, motion parallax is generally considered to be one of the most important depth cues. In fact, it is possible to build three dimensional display systems which solely or in great parts rely on it (see also [11][12][13]). In general, motion parallax is the effect of changing the perspective of a scene according to the movement of its beholder, i.e. the possibility to go around a scene and look at it from different perspectives. In this paper however, it means the changing of the perspective according to the angle the phone is tilted by the user. This is in line with some of the first papers on this effect. According to [11], Herschel stated already 1833 that the perceived motion could either be attributed to the observer or the observed object. A more general definition of motion parallax could therefore be the change of the perspective in accordance to the occurring movement. Normally headtracking should be used to implement this effect, but unfortunately the iPhone does not include a camera which could be used for that.

However, the iPhone (like many other contemporary phones) includes an accelerometer which (among other things) can be used as an input device for computer games (see also [10] on this topic). Moreover, its resolution and accuracy are sufficient to implement a motion parallax effect as well. Normally the perspective change should be calculated using a translation, however this proved to be not feasible in realtime. Through empirical tests it was found though that a simple geometric function with the z-coordinate of the point and the measured value from the accelerometer as inputs works nearly as well to calculate how a point should be shifted to introduce the motion parallax effect. This shifting was implemented in both the x and the y direction. In fact, the implementation was so easy that it is surprising that very few games include such an effect.

Figure 2b) shows how the image looks like after the user tilted the phone to the left. Note that this image has also been distorted in a way similar to using the stereoscopic filter mask (as explained later).

B. Autostereoscopy

It was decided early in the project not to use the software development kit (SDK) provided by the manufacturer of the sheet, since it was too inflexible for this project. However, it was used to get a rough estimate for a filter mask to divide the pixels between the two views.

This mask was refined to one which more resembled the exact sheet that was used during the project. In the end two different masks were implemented, one with an optimal resolution i.e. which divided all pixels evenly between the two views, and one optimized for crosstalk, i.e. which set a few pixel to black which lay exactly between both views. Crosstalk describes the effect if a pixel which should belong to one view influences the other view as well.

These masks had to be realized as a texture (see below). The graphic chip of the iPhone only accepts textures that have a size which is a power of two. However, since the mask is irregular (due to the tiling of the lenticules and their size which cannot be expressed in an integer number of pixels), no resolution could be found that both implemented the mask perfectly and had a size that is a power of two. Therefore, the size of the mask had to be set to the lowest power of two that is bigger than the size of the scene. Since the scene has a size of 320 * 380 pixels, the size of the filter mask became 512 * 512 pixels. Figure 4 shows a magnified patch of one of the used masks with an original size of 32 * 32 pixels. This mask was used during the experiment and will be called “resolution optimized” in the following. The crosstalk optimized mask is more complicated – there, a third color is introduced, which is constantly black, i.e. the pixels with this color are set to black rather than to the color of the corresponding pixel in the left or the right view. This is done to remove pixels which can be equally seen by both eyes of the beholder, thus decreasing crosstalk.

Using an autostereoscopic screen as small as the one on a mobile phone has the disadvantage that the perceived maximal depth is smaller than the one achievable on a bigger screen. However, it has the advantage of being a more controlled environment. For instance, since the phone is held at approximately the same distance by most users, it is easier to find a place where the “sweet spot” should be, i.e. the point where the image quality and the stereo effect are optimal. Unfortunately, the optimal viewing distance of the Wazabee 3Dee Shell seems to be a bit more far away from the screen as a normal usage of the iPhone 3G would suggest.

Another advantage is that it is possible to focus on a single user, i.e. that the generation of 2 images (one for each eye) simultaneously should be enough. Multiview lenticular sheets, which would decrease the resolution enormously, could therefore be avoided. This also allows the usage of user tracking to widen the possible viewing angle. Normally this is done by moving the filter mask (see [16] for an example), which is not possible for the system used. But instead, pixels can be dynamically assigned by the software to either view. For instance, in figure 2 pixel P2 belongs to the right image in 2a), but to the left one in 2b). Knowing the position of the eyes of the beholder and the angle the phone is held in, it becomes possible to adjust the filter mask accordingly. Again, due to the lack of camera the motion accelerator was used (assuming that the distance to the beholder is constant and thus the angle the phone is hold in is the only variable), and a quite simple approximation instead of a computational more expansive full translation. This is called tilt compensation in the following.

As already pointed out, the graphic chip on the iPhone 3G only supports OpenGL ES 1.1, i.e. a static graphics pipeline. The limited resources meant also that it was not possible to implement subpixel resolution for the filter mask and the tilt compensation, as is normally done when rendering for an autostereoscopic display. In this article, subpixel resolution means the segmentation of each pixel in its three different color channels (red, green, blue). Since each pixel consists of different LEDs with different color, it can happen that one or more colors of a pixel should be associated with one view, while the others should be associated with the other view (see also figure 3 and [3]). A lot of papers have been published on the topic on how these subpixels should be distributed among the different views. However, such elaborate algorithms are not feasible on the chosen hardware. Also, anti-aliasing, which normally should be applied on each view separately, is out of reach. (See [17] and [18] for examples on how these functions are normally implemented.) Furthermore, since the mask used has a quite irregular structure (due to the 30 degrees angle of the lenticules, see also figures 3 and 4), artifacts were introduced into the images. The view presented to each eye looks therefore more like the one in Figure 2b) than the original in 2a).

As already described, the actual multiplexing of the views is done using multiple textures. The left view is rendered to a framebuffer instead of to the screen. Then the right view is rendered to the screen, and the left view is rendered to a rectangle the size of the screen. While the left view is rendered, it is multiplied with the mask, thus becoming transparent at the places where the right view should be displayed (see also figure 3). This is a well known procedure for producing partly or completely transparent images in computer graphics and described for instance in [23].

Two important optimizations have been introduced compared to the solution supplied from the manufacturer of the lenticular sheet:

1. The right view is rendered directly to the screen, instead of to another framebuffer, thus saving on memory and computational overhead.

2. The right view is only rendered in places where two different view exists. This is done using the OpenGL scissor functionality, which simply allows no operation outside a predefined area. Thus, the part of the screen which is not covered by the lenticular sheet is rendered only once. This could also be used for objects which do not vary between

the different views, i.e. are placed directly on the screen (instead of behind or in front of it), like e.g. the display of scores, times etc.

Note that it is not possible to simply adjust the size of the rectangle used to render the left view, since it has to be the same size as the view itself. The view is treated as a texture in this case and it would therefore be scaled to match the size of the rectangle. The whole graphics pipeline as used in this project is shown in figure 5.

Apart from being not exactly aligned to the iPhone's screen, the 3Dee Shell has also the drawback that it might sit differently every time it is attached. Therefore the final program should include the possibility to adapt the mask manually to the current position of the lenticular sheet. It is peculiar that such a calibration program is not delivered by the manufacturer of the lenticular sheet, not even in the application they provide for their 3Dee Shell. Without such a calibration, misaligned lenticules may introduces highly visible artifacts like e.g. ghosting.

Three different configuration parameters were included: mask position, view distance and depth. Mask position was used to align the used mask with the current position of the sheet. Since the texture used for the mask is bigger than the screen size, it is possible to let it start at different points, thus allowing this calibration. The sheet may also sit at slightly different angles. However, in these cases it is easier to adjust the sheet instead of trying to turn the mask, because otherwise several textures showing the mask at different angles would have been needed.

View distance describes the distance between the two views. It is included to compensate different eye distances as well as (slightly) different distances in which the phone might be held. However, it was found out that this parameter had little effect on the overall image quality and the stereovision effect.

Finally, depth was included since it was found in [5] that being able to control the level of depth and to switch it off completely is one of the key features that customers expect from a three dimensional display system.

V. Evaluation

Our evaluation procedure was based on the recommendations on assessment methods for multimedia applications [19]. However, a few changes had to be applied due to some practical reasons and to adapt the experiment to the system. The exact setup will be described in the following.

The test was divided in three parts: part 1 dealt with comparison of the two different filter masks, part 2 with a comparison of different 3D solutions and part 3 concluded with a short questionnaire.

Before each test which utilized the autostereoscopic extension of the phone the subject was asked to adjust the mask position so that the subject could see the scene at the best possible image quality and with the best possible stereoscopic effect. It was chosen not to let the subject choose the depth or the view distance since then the setup phase would have become too complicated. Furthermore, it made the tests a little more comparable (since different subjects may otherwise e.g. chose different depths), and a few pre-experiments showed that only the calibration of the mask position is actually needed to allow each subject to configure the phone to get an overall optimal image quality and stereovision effect.

The room in which the experiments were made was held at low illumination to allow the subjects to better see the image on the iPhone. The brightness of the phone was set to maximum for the test as well. The exact light levels in the room or the phone were not measured, but judged to be comparable in all experiments. Although the brightness was lowered by the lenticular sheet in the parts which used it, the difference was barely noticeable.

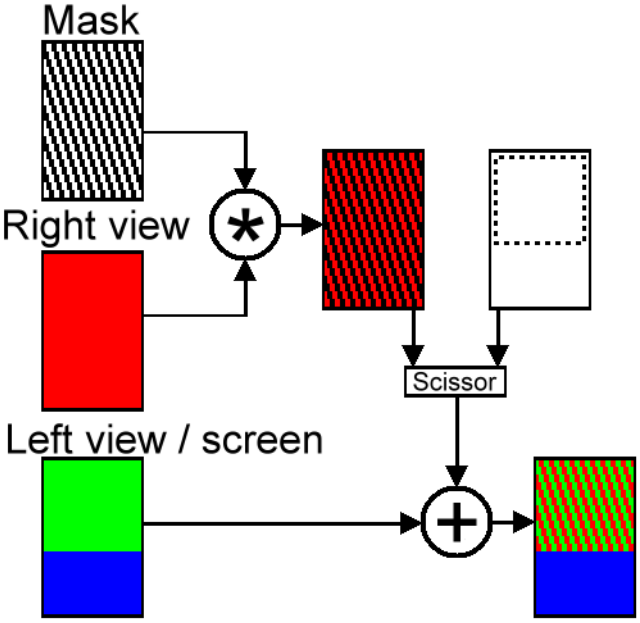

Figure 5. Overview of the graphic pipeline used to implement the autostereoscopic effect

A. Part I: Filter Masks

This test was designed using the pair comparison method described by the ITU [19]. The subject was shown the animated scene described in section 4, rendered using the autostereoscopic sheet and one of the two filter masks. The scene was shown for 10 seconds. Then, after a 2 second break, the subject was shown the same scene but rendered using the other mask, again for 10 s. Then the subject had 10 s to evaluate the masks in three different aspects: image quality, stereo vision effect, and stress factor, i.e. which of the two images was easier to look at.

Figure 4 shows a patch of one of these masks, which evenly distributes the pixels of the display between the two different views. The other mask set a few chosen pixels constantly to zero, since these could be equally seen by both eyes. This was done to remove crosstalk, hence this mask is called “crosstalk optimized” in the following, while the first mentioned mask is called “resolution optimized” since it makes use of all pixels.

The test was replicated three times with each subject to be able to remove insubject variation. The subject was not told in advance which mask were which, not even the number of used masks or that the test would be repeated several times. Which mask was shown first in each sequence was determined randomly before each test.

The subject was asked to hold the phone as still as possible in order not to change the viewing angle and distance during these test sequences, which otherwise could introduce additional crosstalk.

B. Part II: 3D Solutions

For practical reasons, this test was divided in two different phases: one using the stereoscopic sheet and one without it. This was done to avoid reattaching and calibrating the sheet several times, although this meant that the order of sequences of the different tests in this round could not be completely randomized. However, each phase was randomized in itself (i.e. in which order the different sequences of each phase were shown was determined by random in advance), and it was also chosen at random with which phase to begin this part.

For this part of the experiment, it was chosen to use an absolute rating of the different sequences. The rating was done in two different categories (image quality and stereovision effect), and the rating scale ranged from 1 (bad) to 9 (excellent). Each sequence was shown 10 s, then the subject had 10 s to do the actual rating. For this test, the subject was told to tilt the phone as well to capture the effects of motion parallax and the tilt compensation for the autostereoscopic sheet as described in section 4.b).

The sequences with the autostereoscopic sheet included all possible combinations: with and without the tilt compensation as well as with and without motion parallax, altogether 4 sequences.

The sequences without the autostereoscopic display included the normal image and an image which was rendered to include the same distortion as are introduced by the filter mask (see also Figure 2b)), both with and without motion parallax (which made 4 sequences as well). The reason to introduce test sequences which were distorted was to be able to make a more objective comparison with the sequences using the lenticular sheet. Since this part already included 8 sequences it was decided to do no replication in this part. The subject was of course not told in advance or during the experiment in which order the sequences were shown.

The autostereoscopic mask used during this part of the experiment was the one which the subject preferred in the first part of the experiment.

C. Part III: Questionnaire

To set the results into a better perspective and get some additional information, a third part was included which consists of a few questions. In the first question the subjects were asked to rate their experiences with 3D displays so far on a scale from 1 (none at all) to 9 (very much). They were then asked if they would buy a device including a display like the 3Dee Shell, and if, what price they would be willing to pay (not more than for a similar device with a 2D display, a little more or a lot more). They were then asked the same questions, but for a device which would include a display with both very good image and 3D quality.

The test then finished with questions on which services the subjects are using on portable devices at the moment respectively would use if they were available, and for which of these they would choose to use a 3D display if possible. The categories were partly chosen based on the results given in [5], plus a few other ones which are widely used in mobile devices nowadays.

As a last point the subjects were given the opportunity to share all other thoughts that they might have about this experiment or 3D displays in general.

VI. Test Results

Altogether 12 subjects participated in the experiment. Although it was attempted to get a varied test group , most of the participants were male students of a technical program. Note that due to the low number of participants no definite conclusion can be drawn from the results. However, tendencies can be seen, which are furthermore in line with the findings by Tran [20], albeit that project concentrated on desktop systems. A comparison with [7] is difficult since that study concentrated on other aspects, like the effect of different disparities of the stereoscopic image.

A. Part I: Filter Masks

The tests of the filter masks was included to see which kind of trade-off was better – less crosstalk or less resolution loss. Alas, in this perspective the result are not that clear. Most people clearly preferred one mask over the other, and over all, the resolution optimized mask fared better, at least in the categories image quality and stress factor. In 3D effect, it is a tie.

However, it is an open question if and how these results may be used for other systems, since the crosstalk optimized mask introduced visible artifacts by discarding pixels at more or less irregular positions (see also Figure 2b)). These artifacts may have reduced both the subjective image quality and risen the stress factor of the images rendered using this mask.

Furthermore, the resolution after the filtering was very low for each view (around 81 dpi), so that omitting a few pixels is more noticeably as if it would be in a system with a higher resolution. In our opinion the result of this test can therefore not be easily projected to other systems.

The results are shown in table 1.

	Optimized for: Crosstalk	**Optimized for: Resolution**
Image quality	15	21
3D effect	18	18
Stress factor	13	23

Table 1. The number of votes (crosstalk vs resolution) for each mask in the three categories. Higher numbers are better.

B. Part II: 3D Solutions

It could be puzzling that the normal and the artificial distorted images vary in 3D effect as well, and that the tests with the lenticular sheet vary not only in 3D effect, but also in image quality. It seems that image quality and the 3D effect positively influence each other. A high image quality rises the perceived 3D effect and vice versa. In fact, it was shown in [24] that the perceived resolution of an autostereoscopic image is higher than the resolution of each of its two composing images. A perceived higher resolution would explain why a three dimensional image would receive a higher subjective quality than its 2D counterpart.

The three following comparisons are of main interest: 1. using the autostereoscopic enhancement vs not using it, 2. using motion parallax vs not using it, and 3. using the tilt compensation as described in 4.C. vs not using it.

For the first one, we compared the distorted image without the lenticular sheet with the image using the autostereoscopic sheet and tilt compensation. This yields an average improvement of the 3D effect of 1.88 (with a standard deviation of 2.25) if using the autostereoscopic enhancement and tilt compensation. Of course, other comparisons could be possible, but this seems to be the most relevant one, since it should minimize the difference in objective image quality between the two sequences.

For the second one, comparing the sequences using motion parallax with the ones which don't use it, an average difference in 3D effect of 2.58 with a standard deviation of 1.92 is received.

It is interesting that motion parallax gives a higher increase in the 3D effect than the autostereoscopic sheet. However, if the issues with the autostereoscopic sheet could be fixed, the difference might be not that high or even the other way around. It seems to be clear though that motion parallax definitely heightens the perceived depth of the image.

Finally, comparing the sequences using the autostereoscopic sheet and tilt compensation to the sequences using only the autostereoscopic sheet gives an improvement of 1.08 in average (standard derivation 1.33) in image quality and of 1.13 in average (standard derivation 1.44) in 3D effect if using tilt compensation, which is only a moderate improvement but might be higher if the overall quality would be better. Furthermore, if headtracking could have been used instead of the accelerometer, the compensation would have been more accurate and thus the image quality might have been improved further.

The average results of this part of the experiment are given in table 2a and 2b, along with their standard deviations.

auto-stereo-scopy	**tilt compensation**	**motion parallax**	**distorted image**	**mean (std. dev.)**
				7.83 (1.11)
		x		7.83 (1.40)
			x	3.67 (2.35)
		x	x	4.08 (1.83)
x				3.75 (1.48)
x		x		4.50 (1.78)
x	x			4.91 (1.62)
x	x	x		5.50 (1.83)

Table 2a). Results from the 3D solutions test (image quality); ratings range from 1 (bad) to 9 (excellent); standard deviations are given in parentheses

auto-stereo-scopy	**tilt compensation**	**motion parallax**	**distorted image**	**mean (std. dev.)**
				4.25 (2.18)
		x		6.58 (1.93)
			x	3.00 (1.86)
		x	x	5.08 (2.19)
x				3.17 (1.19)
x		x		4.58 (1.88)
x	x			6.42 (1.08)
x	x	x		7.25 (1.14)

Table 2b). Results from the 3D solutions test (3D effect); ratings range from 1 (bad) to 9 (excellent); standard deviations are given in parentheses

C. Part III: Questionnaire

Not surprisingly, most subjects had only limited experiences with 3D systems, with the exception of three which rated their experience level at 5 or 6. Most rated the system used in the test quite low and would not be willing to pay extra for it. However, four out of the twelve subjects would be willing to pay a little more if the quality of the image and the 3D effect would be better, and 3 would even pay a lot more for it. Two of the subjects wouldn’t buy a portable system with a 3D display at all, and the remaining 3 would buy it if wouldn’t cost more than a comparable device with a 2D display. This shows a general interest in portable 3D devices, at least among the participants of the experiment.

An overview of the applications is given in table 3. Interestingly enough, not many would want to watch 3D video on their mobile devices, whether it be movies, series, news, documentations, live stream from events or any form of TV. However, gaming fared very well in comparison. It reached nearly 100% more acceptance as 3D application than the next highest ranking applications. In fact, eight of the twelve subjects (100% of the ones which are currently using their mobile devices for gaming or would do so if their device was capable of that) would choose a 3D display system if they had the choice. A similar high interest in mobile gaming using a 3D display was already reported in [7], which also showed an overall high interest in 3D displays.

Other interesting facts are that the subjects also showed a comparably high interest in stereoscopic versions of location based services, and that more of them would use videophone if it were available in 3D.

The comments given by the subjects where mainly that the system at hand suffered from low resolution, and that the motion parallax effect / the tilt compensation should have a finer resolution. Both issues are shortcomings of the used hardware. Some of the participants remarked that they like glasses-based autostereoscopic solution better, while others would prefer glasses-free solution as in this test. One participant complaint that the configuration of the system was to complicated.

Application	**use / would use in 2d**	**would use in 3D**
games	9	9
taking pictures / movies	12	5
live streams from events	3	2
watching movies / TV series	8	4
watching documentaries / news	6	1
TV (other)	5	1
videophone	4	5
location based services	5	4
social networks	5	0

Table 3. Results from the applications questionnaire

VII. Conclusion

In this paper a 3D system for mobile gaming purposes was presented. The implementation of autostereoscopy and motion parallax used in this system were described, as well as different solutions which had been found to overcome obstacles imposed by the used system.

Furthermore, an experiment has been conducted were several subjects assessed the system and its different possible implementations.

Overall, the subjects showed interest in the system, but also pointed out its shortcomings. Their answers also indicate that gaming will be the most used application for portable devices with 3D displays. All participants which were playing mobile games (or would if they had a mobile phone allowing for that) would prefer to do that in 3D. By comparison, less than half would use it to watch 3D videos. Combined with the fact that 3D could be introduced comparably easily into most games since an extensive 3D description of the gaming world already exists, computer games using 3D displays should gain more attention in the near future.

It was further shown that a very high 3D effect can already be achieved by the usage of motion parallax, which can be introduced quite easily in most modern phones. Furthermore, motion parallax can even be meaningful for the game-play, examples are adventure game where the player has to literally look around the room to find hidden clues or objects, or first-person action games where the player can look around a corner to find out if any dangers lay ahead. We propose therefore that more games capitalize on this effect.

The next step for research in this area might be to implement a system overcoming the shortcomings of the system used in this paper, including a higher resolution display, an OpenGL ES 2.0 capable graphics accelerator and a front-facing camera which can be used for headtracking, thus allowing for a more accurate determination of the user position than if using a motion accelerator. State-of-the-art mobile phones could provide for that. This system could then be used to conduct a wider user study which hopefully would provide more accurate results.

References

[1] K. Willner, K. Ugur, M. Salmimaa, A. Hallapuro, and J. Lainema. Mobile 3D Video Using MVC and N800 Internet Tablet. *Proceedings of the 3DTV Conference 2008: The True Vision - Capture, Transmission and Display of 3D Video*, pp. 69 - 72, 2008.

[2] S. Shi, K. Nahrstedt, and R. H. Campbell. View-dependent real-time 3d video compression for mobile devices. *Proceeding of the 16th ACM international conference on Multimedia*, pp. 781-784, 2008.

[3] J. Konrad, and M. Halle. 3-D displays and signal processing: an answer to 3-D ills? *IEEE Signal Processing Magazine*, 24(6): 97 - 111, 2007.

[4] W. Matusik, and H. Pfister. 3D TV: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes. *Proceedings of ACM SIGGRAPH - International Conference on Computer Graphics and Interactive Techniques*, pp. 814 - 824, 2004.

[5] S. Jumisko-Pyykkö, M. Weitzel, and D. Strohmeier. Designing for user experience: what to expect from mobile 3d TV and video? *ACM International Conference Proceeding Series*, Vol. 291, pp. 183-192, 2008.

[6] R. Cheng, and K. Nahrstedt. Empirical study of 3D video source coding for autostereoscopic displays. *Proceedings of the 15th International Conference on Multimedia*, pp. 573 - 576, 2007.

[7] J. Häkkinen, M. Liinasuo, J. Takatalo, and G. Nyman. Visual comfort with mobile stereoscopic gaming. *Proceedings of the SPIE*, 6055: 85-93, 2006.

[8] M. Callow, P. Beardow, and D. Brittain. Big Games, Small Screens. *Queue*, Vol. 5(7): 40-50, 2007.

[9] D. Nadalutti, L. Chittaro, and F. Buttussi. Rendering of X3D Content on Mobile devices with OpenGL ES. *Proceedings of the 11th International Conference on 3D Web Technology*, pp. 19-26, 2006.

[10] F. Chehimi, and P. Coulton. Motion controlled mobile 3D multiplayer gaming. *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology*, pp. 267-270, 2008.

[11] Ono, H. Perception of Depth, Motion, and Stability with Motion Parallax. *Proceedings of the Second International Symposium on Universal Communication*, pp. 193 - 198, 2008.

[12] T. Suenaga, Y. Matsumoto, and T. Ogasawara. 3D display based on motion parallax using non-contact 3D measurement of head position. *Proceedings of the 17th Australia Conference on Computer-Human Interaction: Citizens Online: Considerations for Today and the Future*, pp. 1-4, 2005.

[13] K. Uehira, M. Suzuki, and T. Abekaw. 3-D Display using Motion Parallax for Extended-Depth Perception. *Proceedings of the IEEE International Conference on Multimedia and Expo 2007*, pp. 1742 - 1745, 2007.

[14] Joel de Vahl. *Intercepting OpenGL calls for rendering on 3D display*. LiTH-ISY-EX-05/3748-SE, 2005.

[15] L.M.J Meesters, W.A. IJsselsteijn, and P.J.H. Seuntiens. A survey of perceptual evaluations and requirements of three-dimensional TV. *IEEE Transactions on Circuits and Systems for Video Technology* 14(3): 381- 391, 2004.

[16] T. Peterka, R.L. Kooima, J. I. Girado, J. Ge, D. J. Sandin, A. Johnson, J. Leigh, J. Schulze, and T. A. DeFanti. Dynallax: Solid State Dynamic Parallax Barrier Autostereoscopic VR Display. *Proceedings of the Virtual Reality Conference 2007*, pp. 155 – 162, 2007.

[17] J. Konrad, and P. Agniel. Subsampling models and anti-alias filters for 3-D automultiscopic displays. *IEEE Transactions on Image Processing*, 15(1): 128 - 140, 2006.

[18] A. Vetro, S. Yea, M. Zwicker, W. Matusik, and H. Pfister. Overview of Multiview Video Coding and Anti-Aliasing for 3D Displays. *Proceedings of the IEEE International Conference on Image Processing 2007*, pp. I-17 - I-20, 2007.

[19] International Telecommunication Union (ITU). Subjective video quality assessment methods for multimedia applications. *ITU-T Recommendation P.910*, 1999.

[20] Yen Tran. *Utvärdering av rörelseparallax i datorspel*. LiTH-ISY-EX—10/4426—SE, 2010.

[21] Jonas Andersson Hultgren. *Metoder för förbättrad rumsuppfattning i körsimulatorer.* LiTH-ISY-EX—11/4442—SE, 2011.

[22] T. Koike, K. Utsugi, M. Oikawa, M. Kobayashi, and M. Yamasaki. Interactive autostereoscopic display with 60 ray directions. *Proceedings of ACM SIGGRAPH - International Conference on Computer Graphics and Interactive Techniques*, Article No. 147, 2006.

[23] T. McReynolds, and D. Blythe. *Advanced Graphics Programming Using OpenGL*. Morgan Kaufmann Publishers / Elsevier, 2005.

[24] Masashi Tsuboi, Shinji Kimura, and Tsutomu Horikoshi. An objective and subjective evaluation of an autostereoscopic 3d display. *Proceedings of the 27th International Conference on Human Factors in Computing Systems*, pp. 3577-3582, 2009.

[25] Wopking, M., 1995. Viewing comfort with stereoscopic pictures: an experimental study on the subjective effects of disparity magnitude and depth of focus. *Journal of the Society for Information Display*, Vol. 3, pp 101-103.

Author Biographies

Jens Ogniewski Born in Helmstedt, Germany, in 1978. He obtained his Master of Science in Electrical Engineering at Linköpings University, Sweden, in 2007 in the field of embedded systems for multimedia and communications. After working at Ericsson he began his Ph.D. Studies at the Image Coding Group at Linköpings University in 2009. His main research interest is parallel computing, especially for graphics and multimedia as well as in embedded systems. He takes a special interest in 3D displays and related technologies.

Ingemar Ragnemalm Born in Linköping, Sweden, 1962. PhD 1993 in image processing at Linköping University at the Department of Electrical Engineering. Worked outside the university 1993-1999, with interactive television and game programming. Teacher and researcher at Linköping University since 1999, teaching computer graphics and game programming. Research interests include image processing, computer graphics, parallel processing and user interfaces.