An Introduction to Augmented Reality

In-Hwan Kim
Department of Electrical and
Computer Engineering
University of Auckland
ikim021@aucklanduni.ac.nz

ABSTRACT

Augmented Reality (AR) is an emerging field in computer vision that has still yet to fully mature. Currently AR systems have been successfully applied in various fields such entertainment, education, rehabilitation and military to name a few. One novel application of AR is in the field of mental and physical rehabilitation. Although similar technologies such as Virtual Reality (VR) have been applied in these fields long before, AR presents several advantages, namely heightened realism and the ability to physically interact with virtual objects.

Recently AR systems on mobile devices have gained popular movement as mobile phones become more and more capable and widespread. However even the fastest mobile phones of today cannot match the computing power of the slowest desktops. The relatively limited computing power of mobile phones causes several problems as AR systems are computationally expensive processes. Despite this, large strides have been made in making mobile AR systems a reality. The proposed solutions vary widely in implementation however they all strive for the same goal – to develop a mobile AR system capable of tracking at 6 degrees of freedom (6DoF) at real time frame rates.

Despite the recent shift towards the mobile AR paradigm, much work still remains for its desktop equivalent. Inefficiencies and grounds for improvement have been identified in the most popular AR toolkit available, ARToolKit [1]. Delays in the rendering pipeline have been identified to produce unfaithful perceptive effects in visuo-haptic AR systems in medical education [2]. Additionally, the inability to scale and track multiple 3D objects based on feature matching is a limitation in the current field of AR [3].

INTRODUCTION

Augmented Reality (AR) is the process in where a live view of the physical world is augmented by computer generated sensory input such as graphics, video and/or sound. AR has been successfully applied in areas such as entertainment, education, rehabilitation and military to name but a few. Traditionally AR systems operate in two distinct phases. The first phase detects interest points (natural features or markers) in the camera images. The second phase establishes a real world co-ordinate system from the data obtained in the first phase. Through these two phases an AR system can then overlay whatever computer generated content is desired over the input images.

In recent times AR on mobile phones has generated vast interest as mobile phones provide the core components required to make AR possible whilst boasting portability and mainstream availability. The aforementioned components are: CPU, display, camera and sensors such as the accelerometer and GPS. However mobile phones offer limited computational power in comparison to desktop computers. Even the fastest mobile phones cannot compare in speed to the slowest desktop computers on the market [4]. This presents challenges for developers of mobile AR applications as efficient utilization of resources is required to run the application at real time frame rates.

Research in physical and mental rehabilitation using AR has also been extensively conducted, often yielding positive results. Systems such as ARcockroach [5] and the AR stroke rehabilitation application by Burke, J.W., et al. [6] has been developed to treat phobia towards cockroaches and physical disability caused by stroke respectively. Although Virtual Reality (VR) systems have already been applied in both areas of study, AR presents several advantages. These advantages largely owe to the fact the patient interacts with a physical environment which is beneficial in order to re-create a certain degree of realism when dealing with phobias and to improve motor skills and muscle strength when dealing with stroke rehabilitation. Furthermore unlike VR, AR requires no specialist hardware and little to no technical expertise to configure and run.

AR has also been applied in the entertainment field such as games. AR games can be classified as serious or casual. The stroke rehabilitation application [6] is an example of a serious game where game principles are applied in order to enhance the enjoyability of the rehabilitation programme. On the other hand, casual games such as Butterfly Effect [7] are aimed at mainstream audiences with no other purpose in mind other than providing entertainment value.

AR is still a relatively new field in computer vision and is yet to fully mature. Many proposals to improve the functionality, performance and efficiency of existing AR toolkits have been suggested. Ruobing, Y. [1] proposes an improvement to the L-k tracking algorithm in ARToolkit, the most commonly used AR development package to increase its performance and efficiency. Youngmin, P., V.

Lepetit, and W. Woontack [3] proposes a scalable method for tracking multiple 3D objects at real time frame rates using natural features whereas existing solutions do not scale well when tracking more than one 3D object. Lastly, Knorlein, B., M. Di Luca, and M. Harders. [2] explores ways to mitigate the effect of visual and haptic feedback delays on the perception of stiffness of virtual objects in visuo-haptic AR systems.

AUGMENTED REALITY ON MOBILE PHONES

In recent times there has been a major shift in the AR space. As mobile phones become more powerful and capable than ever before, more and more AR systems are being developed for the mobile paradigm in comparison to desktops. Despite the appeal of the mobile platform for AR systems, notably widespread availability and low cost it comes with its limitations. These limitations are processing speed, memory size and bandwidth. This has led to an influx of time and money being invested into making AR on the mobile phone practical.

Mobile vs. Desktop Platform for Augmented Reality

Even the fastest mobile phones on the market cannot match the computational power of the slowest desktop. The difference in capability between mobile phones and desktops is at least one order of magnitude and it is likely to stay that way in the foreseeable future [8].

In comparison to desktop CPU's, mobile phone CPU's do not have floating point units (FPU's). This makes floating point calculations costly as the compiler must emulate floating point calculations whenever they are used. As a consequence, floating point calculations are approximately forty times slower than their integer counterparts [8]. Other more obvious shortcomings of the mobile phone CPU are the lower clock speeds and number of cores due to the restrictions owing to factors such as small die size, thermal output and power requirements.

After the CPU, the next most limiting factor in terms of computational power for mobile phones is memory size and bandwidth. Mobile phones of today are designed with a unified memory architecture. This means components such as the CPU, GPU and camera share a single pool of general purpose memory making memory bandwidth scarce [8]. To conserve battery life, not only is the amount of memory limited but it is also slow. This means cache misses are even more costly than on a desktop [8].

The low computational power of mobile phones naturally raises the question of outsourcing computationally intensive tasks to a more capable machine over a network. However, unlike desktops the portability of mobile phones coupled with imperfect network coverage means network connectivity is the exception instead of the norm [8]. Therefore it is impractical to solely depend on network connectivity to run mobile AR applications.

Optimizing Augmented Reality Systems for the Mobile Phone

To efficiently utilize the limited resources of mobile phones it is clear different architectural decisions and algorithms must be used than in desktops [4]. When porting desktop AR systems to a mobile platform this often means a complete re-design of an already existing solution [4]. When designing a mobile AR application from scratch a developer must consider the aspect of software design. Modular based software design is a well-established method to develop large software applications. Breaking down an application into modules with clearly defined interfaces reduces complexity and aids in parallel development. However each module contains a components complete implementation, some of which may be unnecessary [4]. This increases an applications binary size and consequently memory and memory bandwidth use. Therefore it is worth contemplating whether the benefits provided by modular based software design is worthwhile in contrast to a monolithic design which could lead to smaller sized applications on a case-by-case basis.

Furthermore, code reuse is another well-acknowledged method in the programming space. Code-reuse is advantageous as it allows a developer to exert a certain degree of confidence the code has been well tested and is working. It also minimizes effort and shortens development time. However, when developing mobile AR applications caution must be exercised. Re-using existing libraries or solutions to form new ones may not be ideal especially if the libraries or solutions were originally intended for the desktop platform [4]. Therefore they may not be optimized for mobile phones and contain many instances of floating point calculations or non-optimized memory usage as an example. To avoid this pitfall, good practice when developing mobile AR applications is to only re-use existing libraries or solutions that were originally intended for the mobile platform. They are likely to have been developed by experts in the field to deliver high performance and robustness [4].

As discussed previously, mobile phone CPU's do not contain FPU's. Therefore floating point calculations are expensive and are approximately forty times slower than integer calculations [8]. To avoid this, the developer must avoid floating point calculations altogether in the interest of performance. Therefore he or she must take care to use integer or fixed point variants at the cost of reduced precision and numeric range [8]. Similarly, in order to reduce memory and memory bandwidth usage it is recommended the developer leverages techniques such as using compact pixel formats such as RGB565 (16 bits) rather than RGB888 (24 bits) or RGBX8888 (32 bits) in order to produce smaller image sizes [8].

Additionally, parallel execution can be exploited to speed up mobile AR applications. Earlier it was stated a typical AR application operates in two phases. Phase one detects interest points or markers in the camera image. The second phase establishes a real world co-ordinate system from the data obtained in the first phase. Although it is true the second phase is dependent on the first phase's outcome, in reality certain sub-phases in the process chain can be relaxed in order to parallelize their execution [4]. These tasks are camera access, network communication and rendering [4]. As camera access and network communication is I/O bound instead of CPU bound, both can run on separate threads. In network communication most of the time is spent waiting for replies and therefore it can run asynchronously. This is also known as interleaving [4]. This is especially useful for mobile AR applications which use network connectivity as a supplementary option to boost performance or enable networked functionality when available. As many current mobile phones have dedicated 3D accelerators for gaming, it is expected that once 3D accelerators become an integrated part of the CPU, rendering tasks can also be assigned to background threads [4].

PhonyFERNS, PhonySIFT and PatchTracker

Natural feature tracking is a complex problem and demands high computational power [9] compared to traditional marker based tracking techniques. Despite this complication, PhonyFERNS, PhonySIFT and PatchTracker [9] which are the components of a fully self-contained mobile AR system is capable of tracking full 6 degrees of freedom (6DoF) at real time frame rates from natural features. To achieve this, lightweight versions of already existing tracking systems, SIFT and FERNS were produced. The resulting tracking systems PhonySIFT, PhonyFERNS and an innovative template based tracker, PatchTracker is combined into a hybrid tracking system whose individual components work together to become more robust and faster than the individual trackers alone.

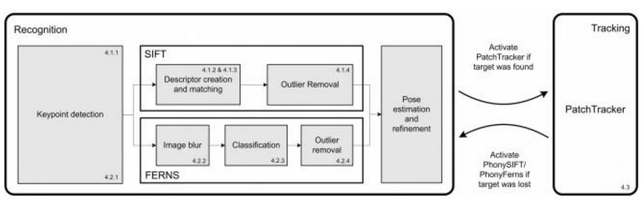

Figure 1. State chart of the PhonySIFT/PhonyFERNS/PatchTracker hybrid system [9].

PhonySIFT differs from SIFT in several ways. The original SIFT uses Difference of Gaussians (DoG) for feature detection whereas this is replaced by FAST in PhonySIFT, known to be one of the fastest feature detectors available [9]. Descriptor creation is achieved using 4x4 sub regions with 8 gradient bins each in most SIFT implementations. In PhonySIFT this is reduced to 3x3 sub regions with 4 gradient bins each. This was found to perform only 10% worse [9] at the gain of performance and optimized memory usage. To match the descriptors created from the camera image to the ones that exist in the database, the original SIFT uses a k-d Tree with the 'Best-Bin-First' strategy. This method was found to be too computationally expensive for scenarios where the descriptors from the camera image vary strongly from the ones in the database [9]. To remedy this, PhonySIFT uses a Spill-Forest consisting of Spill-Trees, a variant of the k-d Tree.

On the other hand, the original FERNS uses the maxima and minima of the Laplace operator to detect interest points in the camera images [9]. Like PhonySIFT, in PhonyFERNS this was replaced with FAST. FERNS database sizes were reduced from 32Mb, which far exceeded available application memory on mobile phones to 2Mb by reducing parameters for Fern sizes [9]. Furthermore, the original FERNS stored matching probabilities as 4-byte floating point values. It was found 8-bit values yielded enough numerical precision to accomplish the task satisfactorily [9] further reducing memory footprint.

PatchTracker is a template based tracker which uses active search compared to tracking by detection used in PhonySIFT and PhonyFERNS. Active search is more efficient than tracking by detection as it exploits the fact that both the scene and pose alter only slightly between two successive frames [9]. Hence the feature positions can be successfully predicted. Therefore PatchTracker compliments the overall system by continuing detection after the initial detection by PhonySIFT and PhonyFERNS in a computationally inexpensive way [9]. Furthermore, PatchTracker addresses the issue of limited tilt angle (40 – 50 degrees) in PhonySIFT and PhonyFerns by continuing to track at close to 90 degree tilt [9].

Evaluation of the overall system produced positive results. The matching rates between the original SIFT and FERNS implementations compared to PhonySIFT and PhonyFERNS were similar [9]. However it was found that drawings and text were not suitable for tracking using PhonySIFT and PhonyFERNS as they typically exhibit repetitive features, high frequencies and few shades of colour which is difficult to uniquely identify [9].

AUGMENTED REALITY APPLICATIONS

AR has produced many novel applications owing to its ability to overlay computer generated input into the real world. This new level of realism has allowed for more effective mental rehabilitation treatments when compared to similar technologies such as VR. On the other hand with the ability to interact with virtual objects in the real world new procedures for physical rehabilitation have emerged. These new methodologies incorporate game principles to stimulate what would otherwise be mundane treatment sessions. Lastly, casual games such as Butterfly Effect [7] are made possible through AR to allow the player to play video games in the third dimension in his or her environment.

An Augmented Reality System for Mental Rehabilitation

ARcockroach [5] is the first AR system for the treatment of cockroach phobias. It was developed using ARToolkit with Virtual Reality Modelling Language (VRML) support. The system was developed with progressive treatment in mind [5]. The therapist can specify how many cockroaches there are, whether they move or not and their size at any given moment. Furthermore the user can kill cockroaches using a flyswatter or insecticide and throw them into a dustbin.

The system was tested with a 26 year old female patient voluntarily seeking help for her psychological disorder. To measure the anxiety levels in the patient a Subjective Units of Distress Scale (SUDS) was used. Before the exposure session, the patient scored a 10 on the SUDS scale when entering a room where a terrarium containing a live cockroach was [5]. After a one hour exposure session with a therapist using ARcockroach [5] the patient was not only able to approach the cockroach in the terrarium, she was able to interact with it and kill it [5].

Figure 2. ARcockroach exposure session [5]

The results of the evaluation were encouraging. Although further evaluation with additional participants is required, it demonstrates AR exposure may be effective in the treatment of these types of phobias. Most importantly it established AR systems such as ARcockroach [5] may be able to induce high levels of patient anxiety similar to that of a real cockroach [5] which is key to its success.

An Augmented Reality System for Physical Rehabilitation

Every year 15 million people suffer from stroke. Of these 15 million individuals, 5 million die and another 5 million are permanently disabled [10]. It has been shown that early, intensive rehab therapy with active functional tasks in an enriched environment leads to positive outcomes [6]. Furthermore victims often report traditional rehabilitation tasks are boring and mundane due to its often repetitive nature [6]. As with ARcockroach [5] it is not the first time technology such as AR has been applied to physical rehabilitation. VR has been applied long before. However VR is inappropriate for home use due to its high cost and level of technical expertise required to set up. AR systems do not exhibit these limitations. Furthermore in an AR system the patients interact with real objects which can result in improved motor skills and strengthened muscles.

Burke, J.W., et al. [6] presents a AR based therapy system with the goal to improve traditional stroke therapy methods to become enjoyable and stimulating. The system incorporates several game techniques in order to achieve these goals. Meaningful play is the relationship between player actions and system feedback. The system uses clear indicators such as coloured rings in its ‘Shelf Stack’ game (see Figure 3) or disappearing bricks in its ‘Brick a Break’ game (see Figure 3). Meaningful play is important as players must be aware of their goals, the actions needed to achieve these goals and whether they are achieving these goals or not in order for effective engagement with the game [6].

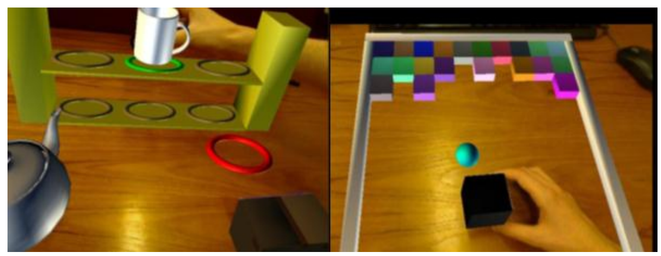

Figure 3. Demonstration of meaningful play [6]

Dynamically adjusting the level of difficulty is vital to a games success. If the game is too easy, it becomes boring. If the game is too hard, it becomes frustrating. On the other hand, effective handling of failure is also important as the intended audience exhibit poor motor controls and unfamiliarity with the game [6]. Furthermore failures must be handled correctly so the patients are not discouraged if they did not perform as well as they had hoped.

Brick’a’Break (see Figure 3, right) is inspired from Atari’s 1976 ‘Breakout’ game. The player must clear rows of bricks at the top of the playing field by rebounding a ball with the paddle controlled using a real world object. Unlike the original game the player does not lose lives if the ball is dropped and there is no time limit to clear the bricks. This abides with the principle to handle failures effectively. However, the end score is determined by how fast the player can clear the bricks to achieve meaningful play.

In Shelf Stack (see Figure 3, left) the player is given several real world objects of differing size, shape and width representing different virtual objects. The player is also presented with virtual shelves with each shelf consisting of rings. During the game an object and shelf ring is selected at random and highlighted with a red circle indicating the designated object and its target. The player must place the chosen object on the indicated ring on the shelf. Upon completion the ring on the shelf turns green.

During the initial evaluation some issues were identified with the system. To complement good hand-eye co-ordination the camera must ideally be placed as close to the eye position of the player as possible [6]. This can become an issue for games that require use of both upper limbs instead of just one. There were also problems with depth perception. In AR systems it can oftentimes be difficult to perceive depth in a scene without use of visual clues such as shadows or motion parallax. Lastly, blocking a marker or having insufficient lighting in a room can cause issues with tracking. This can be offset with more robust tracking algorithms.

Augmented Reality Systems for Home Entertainment

Butterfly Effect [7] is a 3D AR puzzle game developed using Designers Augmented Reality Toolkit (DART) with mainstream audiences on next-generation video game consoles in mind. The key motivation in Butterfly Effect [7] was to create a game that leverages the structure of the players environment without having the system require a model of the physical space [7]. This is important if the game is to become mainstream as it is unlikely AR games such as Butterfly Effect [7] will become feasible for home audiences if the system requires a detailed and accurate model of the physical location it is played in.

The player is presented with a distributed collection of virtual butterflies in a 3D volume around him/her. The game is played in any physical location the player wishes. The novelty of Butterfly Effect [7] is that an instance played in one location differs from another. For example, an instance played in a home environment compared to one played in the backyard. The player travels his or her environment capturing the virtual butterflies that surround him/her. Butterflies are captured by getting close to them. For butterflies out of reach, the player can rotate the virtual space in 90 degree portions around a player specified axis using the 'Tornado Stick' (see Figure 4) to bring them to an accessible location. The key challenge is for the player to capture all the butterflies scattered in his or her environment.

Butterfly Effect [7] has been designed to take practical constraints into account. As quick, abrupt movement is generally harder to track than slow, thoughtful ones Butterfly Effect [7] encourages slow and thoughtful movement as part of its game design [7] when compared to games such as a first person shooter. Not requiring a model of the physical space makes it difficult to adjust the level of difficulty in the game. The authors are currently uncertain of a solution to this problem [7]. As with Shelf Stack [6] and Brick'a'Break [6], depth perception becomes an issue. Although this issue is somewhat mitigated by the use of motion parallax and object size adjustment depending on its distance, the current approach is to include it into Butterfly Effect as an acquirable skill [7].

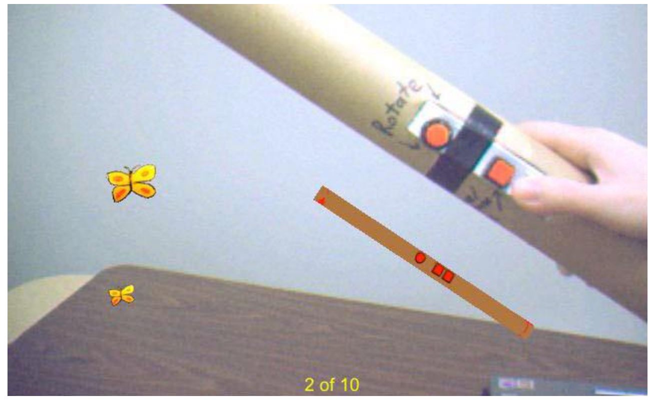

Figure 4. Tornado Stick and virtual butterflies [7]

IMPROVEMENTS TO TRADITIONAL AUGMENTED REALITY APPROACHES

AR is a relatively new field in computer research and has numerous limitations and inadequacies. Several of these have been identified and improved upon. Ruobing, Y. [1] claims traditional methods of registration and tracking in AR systems are inaccurate and inefficient [1] and proposes a new and improved method. Youngmin, P., V. Lepetit, and W. Woontack. [3] identifies a weakness in multiple 3D object tracking using natural features in current AR systems despite the clear need for a scalable and efficient solution [3]. Lastly, Knorlein, B., M. Di Luca, and M. Harders. [2] explores the effect of delays in visual and haptic feedback on the perception of stiffness of a virtual object.

Improved Registration and Tracking for ARToolKit

Current methods of registration and tracking in AR systems are inaccurate and inefficient [1]. Ruobing, Y. [1] presents an improved registration and tracking algorithm based on feature matching which significantly improves the robustness and efficiency of registration and tracking. This was achieved by improving the traditional L-k tracking algorithm used in ARToolKit, the most popular AR development kit currently available [1]. ARToolKit uses a simple template matching method and therefore suffers from high false recognition rates [1]. The new proposed method uses feature matching and has made several changes to the existing L-k tracking algorithm. The L-k tracking algorithm's basic purpose is to consider all the pixels in two given images and compute their similarity levels. To achieve this, the Hessian matrix is calculated every iteration in the original algorithm. The new proposed method calculates this value once in advance and updates it every iteration using update formulas which are significantly less computationally intensive [1]. Experimental results show the new L-k algorithm is robust and accurate and has better real time performance than the original implementation [1].

Multiple 3D Object Tracking for Augmented Reality Systems

Existing methods for 3D object tracking do not scale well with an increasing number of objects, despite the need for an effective solution for applications such as tangible interfaces and table-top AR [3]. Youngmin, P., V. Lepetit, and W. Woontack. [3] presents a solution that is able to track several 3D objects simultaneously in real time. This method combines object detection and object tracking and exploits the advantages of both approaches. Frame by frame object detection is robust but slow whereas object tracking is less computationally demanding but more prone to failure [3]. The system performs object detection and tracking on two separate cores, leveraging multi-core CPU's of today [3]. To improve efficiency, the computational complexity of detecting multiple 3D objects is spread over consecutive frames [3]. In other terms, the system does not try and detect every object in a frame at once but only as much as possible to maintain real time frame rates. Therefore objects not detected in one frame will be detected in the next few frames. This results in a small delay that is kept under 1 second in practice so is not easily perceptible to users [3].

Once an object is detected the system initializes frame by frame tracking for the object. Tracking is performed by relying on interest points detected on the objects surface. Therefore it can continue to estimate object pose after the initial detection. Throughout this the object detection module continues its attempt in detecting objects even when object tracking is running for them to prevent common problems such as loss of tracking due to abrupt motion [3]. This was only feasible as both the object detection and object tracking module were run on separate cores.

Visual and Haptic Delays and Stiffness Perception

With current technology visual delays in AR systems are unavoidable and can occur in different steps of the rendering pipeline [2]. It has been shown haptic feedback delays resulted in a decreased perceived stiffness whereas visual feedback delays caused an increased perceived stiffness [2]. Accurate representations in visual and haptic properties of deformable objects is vital in the application of visuo-haptic AR in medical education [2].

Visual delays can occur in three steps of the pipeline: data acquisition, data processing and during the display [2]. Haptic delays mainly occur during data processing while data acquisition and display delays can be assumed to be zero [2]. During the evaluation a virtual spring was realized in the visuo-haptic AR environment (see Figure 5). Experimental conditions were created by introducing artificial delays in different stages of the rendering pipeline [2]. Fourteen participants took part in the evaluation and it was observed simultaneous delays of both visual and haptic properties led to a partial compensation these effects [2].

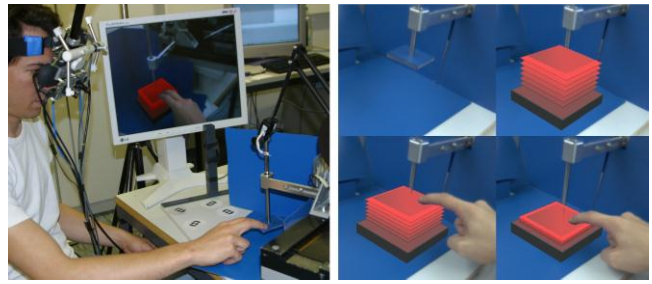

Figure 5. Visuo-haptic AR setup and the virtual spring [2]

SUMMARY

Due to the limited computing power of mobile phones new and innovative ways to implement AR systems for the mobile paradigm have been proposed. Approaches vary widely, from producing lightweight version of already existing desktop AR tools such as SIFT and FERNS [9] to developing specialized AR systems from the ground up optimized for the architecture and limitations of mobile devices [4, 8]. The two main reoccurring themes however are either producing highly efficient and optimized mobile AR applications [1, 4, 8] or developing a hybrid system that uses two or more approaches for registration and tracking and exploiting their respective qualities [3, 9]. Above all, one fact is clear – mobile AR systems require a whole new approach in design and implementation and it is not sufficient to simply port desktop AR applications to mobile devices [4].

AR has demonstrated its effectiveness in mental and physical rehabilitation [5, 6]. Although VR has been used in the respective fields before, AR presents several advantages namely heightened realism and the ability to physically interact with virtual objects. This is crucial for mental rehabilitation such as phobia treatment to induce similar levels of anxiety as the patient would experience in real circumstances [5]. The ability to interact with physical objects is important for physical rehabilitation such as stroke treatment so patients can improve their motor skills and strengthen muscles [6].

AR is still a relatively new field of research in computer vision and thus comes with its inadequacies. Many grounds for improvement in existing solutions have been identified and studied. These include more efficient and robust tracking algorithms in ARToolKit [1], a way to track multiple 3D objects from natural features at real time frame rates [3] and methods to compensate for delays in the rendering pipeline of visuo-haptic AR systems resulting in unfaithful perceptions of stiffness [2].

FUTURE WORK

Mobile phones appear to be the next frontier in AR development. As faster and more capable mobile phones are produced every year, an area that is still yet to mature is AR task parallelism in multi-core phones. With more resourceful use of the computing power offered by today's mobile phones, faster and more robust AR systems can be developed on the mobile platform that may eventually match its desktop equivalents in terms of performance.

The limited screen real-estate in mobile phones can quickly become an issue especially when there is a lot of computer generated content to overlay in the input images. This problem could be resolved by using digital glasses combined with a camera. The glasses could be powered by connecting it to a mobile phone inside an individual's pocket. This will allow it to utilize the computing power of the mobile phone to render images and its associated AR content in the heads up display (HUD).

REFERENCES

1. Ruobing, Y. The Study and Improvement of Augmented Reality Based on Feature Matching. In *Software Engineering and Service Science (ICSESS)*, 2011 IEEE 2nd International Conference on. 2011, 1-4. http://ieeexplore.ieee.org.ezproxy.auckland.ac.nz/stamp/stamp.jsp?tp=&arnumber=5982388
2. Knorlein, B., M. Di Luca, and M. Harders. Influence of Visual and Haptic Delays on Stiffness Perception in Augmented Reality. In *Mixed and Augmented Reality, 2009.* ISMAR 2009. 8th IEEE International Symposium on. 2009, 1-4. http://ieeexplore.ieee.org.ezproxy.auckland.ac.nz/stamp/stamp.jsp?tp=&arnumber=5336501
3. Youngmin, P., V. Lepetit, and W. Woontack. Multiple 3D Object Tracking for Augmented Reality. In *Mixed and Augmented Reality, 2008.* ISMAR 2008. 7th IEEE/ACM International Symposium on. 2008, 1-4. http://delivery.acm.org.ezproxy.auckland.ac.nz/10.1145/1610000/1605357/04637336.pdf?ip=130.216.15.64&acc=ACTIVE%20SERVICE&CFID=76873094&CFTOKEN=77306005&__acm__=1334136787_4773700c5bebbbe2e67bd517aa1730b4
4. Wagner, D. and D. Schmalstieg, Making Augmented Reality Practical on Mobile Phones, Part 1. *Computer Graphics and Applications*, IEEE, 2009. 29(3), 12-15. http://ieeexplore.ieee.org.ezproxy.auckland.ac.nz/stamp/stamp.jsp?tp=&arnumber=4909113
5. Juan, M. C., Botella, C., Alcaniz, M., Banos, R., Carrion, C., Melero, M. and Lozano, J. A. An Augmented Reality System for treating psychological disorders: Application to phobia to cockroaches. In *Mixed and Augmented Reality, 2004.* ISMAR 2004. Third IEEE and ACM International Symposium on. 2004, 1-2. http://ieeexplore.ieee.org.ezproxy.auckland.ac.nz/stamp/stamp.jsp?tp=&arnumber=1383064
6. Burke, J. W., McNeill, M. D. J., Charles, D. K., Morrow, P. J., Crosbie, J. H., McDonough, S. M. Augmented Reality Games for Upper-Limb Stroke Rehabilitation. In *Games and Virtual Worlds for Serious Applications (VS-GAMES)*, 2010 Second International Conference on. 2010, 1-4. http://ieeexplore.ieee.org.ezproxy.auckland.ac.nz/stamp/stamp.jsp?tp=&arnumber=5460103
7. Norton, M. and B. MacIntyre, Butterfly Effect: An Augmented Reality Puzzle Game. In *Proceedings of the 4th IEEE/ACM International Symposium on Mixed and Augmented Reality*. 2005, IEEE Computer Society, 212-213. http://delivery.acm.org.ezproxy.auckland.ac.nz/10.1145/1110000/1105217/24590212.pdf?ip=130.216.15.64&acc=ACTIVE%20SERVICE&CFID=76873094&CFTOKEN=77306005&__acm__=1334136824_28d6615d76a5587166b4941f8410bda7
8. Wagner, D. and D. Schmalstieg, Making Augmented Reality Practical on Mobile Phones, Part 2. *Computer Graphics and Applications*, IEEE, 2009. 29(4), 6-9. http://ieeexplore.ieee.org.ezproxy.auckland.ac.nz/stamp/stamp.jsp?tp=&arnumber=5167481
9. Wagner, D., Reitmayr, G., Mulloni, A., Drummond, T., Schmalstieg, D. Real-Time Detection and Tracking for Augmented Reality on Mobile Phones. *Visualization and Computer Graphics*, IEEE Transactions on, 2010. 16(3), 355-368. http://ieeexplore.ieee.org.ezproxy.auckland.ac.nz/stamp/stamp.jsp?tp=&arnumber=5226627
10. Stroke Statistics. http://www.strokecenter.org/patients/about-stroke/stroke-statistics/.