

April 2017

Using Image Recognition Technology as a Gameplay Mechanic

Andrew Laurence Ray
Worcester Polytechnic Institute

Thomas James Meehan
Worcester Polytechnic Institute

Yahel Nachum
Worcester Polytechnic Institute

Follow this and additional works at: <https://digitalcommons.wpi.edu/mqp-all>

Repository Citation

Ray, A. L., Meehan, T. J., & Nachum, Y. (2017). *Using Image Recognition Technology as a Gameplay Mechanic*. Retrieved from <https://digitalcommons.wpi.edu/mqp-all/922>

This Unrestricted is brought to you for free and open access by the Major Qualifying Projects at Digital WPI. It has been accepted for inclusion in Major Qualifying Projects (All Years) by an authorized administrator of Digital WPI. For more information, please contact digitalwpi@wpi.edu.



WPI

MQP MBJ-1700

Using Image Recognition Technology as a Gameplay Mechanic

A Major Qualifying Project
Submitted to the Faculty of the
WORCESTER POLYTECHNIC INSTITUTE
in partial fulfillment of the requirements for the
degrees of BACHELOR OF SCIENCE
in COMPUTER SCIENCE and
INTERACTIVE MEDIA AND GAME DEVELOPMENT

Submitted by:

THOMAS J. MEEHAN
YAHIEL NACHUM
ANDREW L. RAY

Advisors:

Prof. Brian Moriarty
Prof. Charles Rich

Submitted:

April 25, 2017

This report represents the work of WPI undergraduate students submitted to the faculty as evidence of completion of a degree requirement. WPI routinely publishes these reports on its website without editorial or peer review. For more information about the projects program at WPI, please see <http://www.wpi.edu/academics/ugradstudies/project-learning.html>

Abstract

Image recognition (IR) has been used in previous entertainment applications, but for various reasons, these applications have had only moderate success. Meanwhile, art museum attendance has been declining in recent years. Our project attempts to demonstrate the feasibility of using IR technology as a gameplay mechanic in an application designed to increase the engagement of art museum attendees. Our mobile scavenger hunt, *The Great Art Quest*, is designed for deployment in any art museum.

Acknowledgements

Many people have helped us with this project. We would like in particular to thank the staff of the Worcester Art Museum (WAM), without whom this project would not have been possible, and especially Prof. Jeffrey Forgeng, our WAM liaison, for allowing us to use their facility as a testing location. We would also like to thank our many playtesters, who volunteered their valuable time and effort to improve the quality of our application. Another person whom we would like to thank is Ravi Parekh, who assisted with the initial research for our project. Finally, we would like to thank our advisors, Profs. Brian Moriarty and Charles Rich, for dedicating their time to ensuring that this project was a success.

Glossary of terms

Accuracy: The detection rate for positive results, the inverse of the rate of false negatives, e.g. if from a set of trials, six should be positive and four of those six actually *are* positive, the accuracy is 67%.

Alpha: The stage of game production in which all expected functionality is implemented, but placeholder assets may still be present.

Application Programming Interface (API): A computer technology that allows a program to interface with another program using a documented library of procedures, functions and data structures.

Artificial Intelligence (AI): Any one of a class of computer technologies that use a decision structure to perform a complex operation.

Asset: Any forward-facing content in a game, including images, sounds, and videos.

Beta: The stage of game production in which all all expected functionality and assets are implemented, though bug-fixing and polishing may still be required.

C#: A high-level, object-oriented programming language based on C and C++.

Deep learning: Any of a class of machine learning algorithms that use a multilayered set of processing units to perform transformation on a set of data and learn to construct multiple levels of representations corresponding to different levels of abstraction [1].

False negative: When a result is returned as negative when it should have been a positive.

False positive: When a result is returned as a positive when it should have been a negative.

HTTP request: A message containing a query sent to a web server asking it for a response.

Image recognition (IR) technology: A computer technology that uses some method of data analysis to determine the contents of a digital image.

JavaScript Object Notation (JSON): A syntax for formatting data in a computer-readable way based on JavaScript's notion of objects and arrays.

Noise: In digital images, random extraneous colorings that result in a grainy appearance, typically resulting from oversensitivity in the camera capturing the data.

POST: A type of HTTP request that sends its query as part of the message body.

Rate limiting: A usage restriction imposed by some web services that prevents users from sending more than a set maximum number of requests within a set time interval.

Specificity: The detection rate for negative results, the inverse of the rate of false positives, e.g. if from a set of ten trials, four should be positive, and six (including those four) are positive, the specificity is 67%.

Training set: In artificial intelligence, a set of input data that is used to adjust the processing model to improve accuracy.

User interface (UI): The part of a piece of software with which the user directly interacts, such as buttons, text segments, and input fields.

Contents

Abstract	i
Acknowledgements	i
Glossary of Terms	ii
1 Introduction	1
2 Background	2
2.1 Image recognition (IR) technology	2
2.1.1 CloudSight	2
2.1.2 ClarifAI	3
2.1.3 IBM Watson image recognition	4
2.2 Pitfalls of IRT	5
2.2.1 The “two giraffes” problem	5
2.2.2 Distinguishing noise	6
2.2.3 Training sets	7
2.3 Public application programming interfaces	7
2.4 Previous work	7
2.4.1 <i>Hwat?!</i>	8
2.4.2 <i>LEGO Fusion</i>	8
2.4.3 <i>CamFind</i>	9
2.5 Problems with previous work	9
3 Design	10
3.1 Planning	10
3.1.1 Exploring public APIs	10
3.1.2 Playability	11
3.2 Game mechanics	12
3.2.1 Scavenger hunt	12
3.2.2 Augmented reality	12
3.2.3 The final puzzle	13
3.2.4 Side mechanics	13
3.3 Graphic design and user interface	14
3.3.1 Parodying free-to-play	14
3.4 Game narrative	14
3.4.1 Initial attempts	14
3.4.2 Final backstory	15
3.4.3 Plot outline	15
3.4.4 Completed narrative	16
4 Development	17
4.1 Process	17
4.1.1 Core mechanic	17

4.1.2	Side panels	18
4.2	Testing	18
4.2.1	Building a testing plan	18
4.2.2	Execution	18
5	Evaluation	19
5.1	Testing results	19
5.1.1	Summary of Round 1	19
5.1.2	Summary of Round 2	20
5.1.3	Summary of Round 3	22
5.2	Capability of IR technology for interactive media	22
5.3	Possible future improvements	22
	Works cited	23
	Appendix A: Game design document	25
	Appendix B: Narrative document	29
	Appendix C: UI design	34
	Appendix D: Testing survey	42
	Appendix F: IRB protocols	43

List of Figures

2.1	CloudSight logo	2
2.2	JSON format data from the CloudSight API, returned in response to a photo of Beats headphones by Dr. Dre.	2
2.3	An example of a “reverse Turing test” image, used to determine if an IR service employs human workers.	3
2.4	ClarifAI logo	3
2.5	JSON data from the ClarifAI API.	4
2.6	Watson’s avatar	4
2.7	JSON data returned by Watson’s IR service API.	5
2.8	Windows 10 logo recognition fails in Microsoft’s own IR service.	6
2.9	Photograph with three levels of noise	6
2.10	<i>Hwat?!</i> app screenshot	8
2.11	<i>LEGO Fusion</i> app screenshot	8
3.1	Distribution of Response Times for the services.	11
3.2	Comparison of the three services considered.	11
3.3	Phone paper prototype	11
5.1	Effect of app on museum engagement, round 1	19
5.2	Percieved image identification accuracy, round 1	20
5.3	Effect of app on museum engagement, round 2	21
5.4	Percieved image identification accuracy, round 2	21

1 Introduction

Image recognition (IR) technology is software designed to analyze digital images and identify their contents. IR has many potential uses. It is often employed in test automation and quality control automation [2]. It also has several applications in the medical industry. A leader in endoscopic technology, i-NSide, has created a smartphone attachment that takes medical-grade images of the human ear, and uses them to form diagnoses [3]. TapTapSee is an accessibility tool for blind people that identifies scenes using IR technology [4].

IR also has applications in the entertainment industry. Several products, such as *Hwat?!*, *LEGO Fusion*, and *CamFind* have demonstrated the use of using automated image classification as a gameplay mechanic, but none so far have demonstrated its viability [5] [6].

There are several services, including CloudSight, ClarifAI, Watson IR, and Google Cloud-Vision, that provide application programming interfaces (APIs) for third-party use [3] [7] [8]. These APIs provide a simple way for academic and commercial developers to make use of this powerful technology. They typically provide a web-based APIs that can be accessed using HTTP requests [7].

The goal of our project was to produce an alpha prototype of a mobile game demonstrating the potential of IR technology as a gameplay mechanic. To that end, we looked at various mobile development environments, finally settling on Unity. Unity is a cross-platform game development environment programmed primarily in C#. It is widely used by both independent and triple-A game developers [9].

We decided to develop a scavenger hunt application for use in art museums. This led us to another goal: demonstrating the ability for game technology to increase public engagement with art. This is important because art museum attendance has been on a downward trend in recent years [10].

Most existing exhibit-engagement programs are specific to one museum. Because we used a generic IR service, the game which our group prototyped can be used in *any* art museum. This is because our game, rather than looking for specific works of art, looks for particular items in those works, which should be discoverable in any metropolitan-scale art collection.

2 Background

In order to achieve our goal, our team did research on several different topics and endeavored to make an informed decision about the several IR APIs that were available for us to use. To determine likely shortcomings in our application, we did research on existing uses of IR technology in games like *Hwat?!* and *LEGO Fusion*.

2.1 Image recognition (IR) technology

Several different APIs are available for IR technology. They each have different benefits and problems, and our group needed to determine which was the most suitable for our needs. Several criteria were used (see Section 3.1.1: Exploring public APIs) to make this determination, based on our research.

2.1.1 CloudSight

CloudSight (Figure 2.1) is a company that offers a public API for IR. The user uploads an image and the API returns a description of it. What sets CloudSight apart from other IR services is that it returns a human-readable phrase describing the image rather than tags of what the image might contain. The API's ability to return an accurate and well-worded description of the image's contents gives the service a peculiarly human-like "feel." For example, in Figure 2.2 the description includes the sentence "red beats by dr. dre wireless headphones." By contrast, competing IR services, such as ClarifAI, would return a simple list of tags such as: sound, stereo, electronics, and speaker.

CloudSight uses deep learning in order to train their IR system [12]. The API is not user-trainable, but its IR service is accurate enough to work in many different situations and environments. Its high performance allows developers to concentrate their development efforts on handling image requests to CloudSight's API rather than implementing their own IR system. [8]



Figure 2.1: CloudSight logo [11]

Figure 2.2: JSON format data from the CloudSight API, returned in response to a photo of Beats headphones by Dr. Dre.

```
{  
  "token": "  
    oyJsaGvShy65RHoRGovAhw",  
  "url": "http://assets.  
    cloudsight.ai/uploads/  
    image_request/image  
    /207/207970/207970368/81  
    TldxYJmL._SL1500_.jpg",  
  "ttl": 53,  
  "status": "completed",  
  "name": "red beats by dr.  
    dre wireless headphones"  
}
```

One of the apps that has integrated the CloudSight API focuses on accessibility. The Tap-TapSee app allows a blind person to take pictures of the world and get spoken descriptions of those pictures read back to them.

There is speculation that the service is actually powered by a “mechanical Turk,” meaning the images are simply fed to human workers to identify them. This would explain the incredible accuracy, extremely high request latency, spelling errors, and human-like oversights that have been reported. Suspicions of the service being a Turk originated with Cloudsight’s own terms of service, which state “Our service employs a proprietary technology that utilizes both computer vision and crowdsourcing” [13]. Consider the image shown in Figure 2.3, which produced the response “I love Paris in the springtime” instead of the text actually supplied, “I love Paris in the the springtime.” This sort of typographic error is something a person might easily commit, but not a machine algorithm. Furthermore, when submitting two identical images under different names, the service often returns substantially different results, suggesting both that the image is submitted for human classification each time, and that filenames are being used to detect duplicate images instead of a hash algorithm like MD5. Finally, descriptions themselves are occasionally prone to typos, suggesting that a person is typing them out on a case-by-case basis.



Figure 2.3: An example of a “reverse Turing test” image, used to determine if an IR service employs human workers.

2.1.2 ClarifAI

ClarifAI (figure 2.4) is one of the leading APIs for image and video recognition. It provides a variety of IR services, including automatic tagging of images (accompanied by probability values), tag-based search, reverse search for similar images, automatic image moderation, and training through customizable or generic models. Generic models include “food,” “celebrities,” “travel,” “weddings,” “face and emotion detection,” and “apparel.” API calls are made by providing a URL link to an image, or by directly uploading the image via an HTTP request. [3]



Figure 2.4: ClarifAI logo. [3]

Real-world applications of ClarifAI include a photography marketplace called Foap that receives thousands of user-supplied photos every day. In order to connect brands with the images they're looking for, Foap uses ClarifAI to consistently tag all these photos, making them easily searchable. The service essentially parses stock images to provide more useful tags.

Another example is Asset Bank, a leading digital asset management solution that provides enterprises with highly-configurable solutions to manage their media libraries. Leveraging image recognition, Asset Bank built an auto-tagging feature into their platform to create a better user experience for their customers. It can even be applied to healthcare. I-Nside, a leader in endoscopic technology, created a smartphone attachment that takes medical-grade images of the human ear. Using over 10,000 ear photos, ClarifAI trained a custom model that accurately identifies common ear problems. This model is used in i-Nside's technology to help diagnose ear diseases all over the world.

Figure 2.5: JSON data from the ClarifAI API.

```

"data": {
  "concepts": [
    {
      "id": "ai_HLmqFqBf",
      "name": "train",
      "app_id": null,
      "value": 0.9989112
    },
    {
      "id": "ai_fvlBqXZR",
      "name": "railway",
      "app_id": null,
      "value": 0.9975532
    }
  ]
}

```

2.1.3 IBM Watson image recognition

Watson (figure 2.6) is a mainframe-based AI created by IBM. Though best known for winning the TV trivia game show *Jeopardy* [14], Watson's algorithms are widely deployed across a wide range of applications. The Watson API, available through Bluemix and the Watson Developer Cloud, offers services such as language translation, natural language classification, text-to-speech, and IR [7].

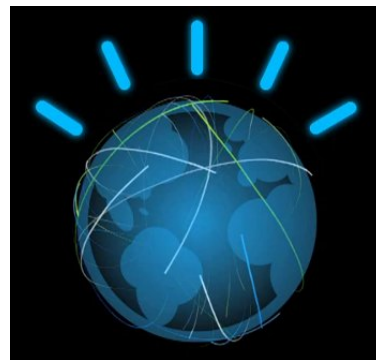


Figure 2.6: Logo for IBM Watson. [7]

The Watson IR service is trainable, meaning that it is possible to build up a database of images, both positives and negatives, which the system will use to analyze future requests. Positive examples are examples of images of the target object, and improve the *accuracy* of the system, reducing the rate of false negatives by introducing more data with which to compare images under inspection. Negative examples are images that do *not* contain the target object, but contain visually similar objects. Negative examples improve the *specificity* of testing, reducing the rate of false positives by providing more data against which images under test may be contrasted. [7]

A generic database is provided for cases for which insufficient time is available to build a customized training set. This database covers many different fields, but is not well-specialized. We used this database to compare the Watson IR service to CloudSight and ClarifAI.

When given an image, Watson IR returns a JavaScript object notation (JSON) dataset containing a list of the classes in which the service thinks the image belongs, together with the confidence interval or “score” for each class. Classes may be grouped into type hierarchies. For example, Figure 2.7 shows the type hierarchy “/appliance/gas oven”, indicating that a gas oven is a subtype of appliance. [7]

2.2 Pitfalls of IRT

While IR technology is already remarkably powerful, and offers a promising future as algorithms are refined and computational power increases, current implementations present a number of significant shortcomings. The “two giraffes” problem, noise effects, and training are three issues that were faced by our group during the project.

2.2.1 The “two giraffes” problem

Many IR systems are plagued by the “two giraffes” problem, a defect which causes the service to return obviously wrong results for no apparent reason. The reasons for this error are twofold: a problematic training set (involving over- or under-representation of essential image categories), and the requirement to return a result from ambiguous analysis.

For example, Google’s Deep Dream AI is notorious for erroneously detecting dog faces in many of the images it inspects. This happens because its AI was trained with an image set incorporating “fine-grained classification[s] of 120 dog sub-classes.” [15] Such lopsided training accounts for Deep Dream’s apparent obsession with dogs.

Figure 2.7: JSON data returned by Watson’s IR service API.

```
[
  {
    "classes": [
      {
        "class": "gas oven",
        "score": 0.872,
        "type_hierarchy":
          "/appliance/gas oven"
      },
      {
        "class": "oven",
        "score": 0.883
      },
      {
        "class": "appliance",
        "score": 0.97
      }
    ]
  }
]
```

This problem also occurs when AIs are asked to classify images with very little resolution. For example, when Microsoft’s IR service was given the Windows 10 logo (Figure 2.8), an image it should presumably be easily capable of identifying, it instead responded with “I think it’s two giraffes in a fenced in area.” Granted, this report was preceded with the qualifier “I am not really confident,” indicating that the AI was aware that its response was unlikely to be correct. Nevertheless, it may be preferable for IR agents to respond to high uncertainty with the virtual equivalent of a shrug rather than risk returning an absurd (and potentially embarrassing) classification.

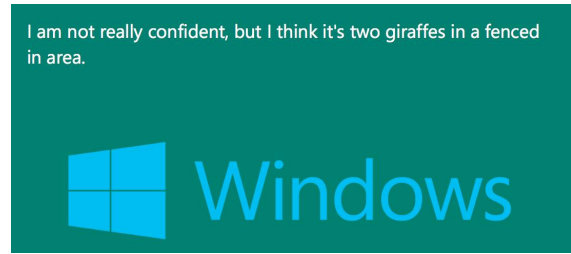


Figure 2.8: Windows 10 logo recognition fails in Microsoft’s own IR service.

2.2.2 Distinguishing noise

Another issue with IR technology is the problem of visual noise. While research has been done to isolate sources of noise and remove it [16], noise can still increase the likelihood of both false positives and false negatives in automated IR systems. In one test involving an image consisting of pure noise, the CloudSight API reported that the image was too blurry to analyze. ClarifAI classified the same image as “wallpaper” and “vintage,” while Watson identified it as “fabric” with a 77% level of confidence. The presence of noise was presumably responsible for these false positives.

False negatives are also possible. Presented with the leftmost image in Figure 2.9, Watson identified a bell tower with 85% certainty and watercraft with 59% certainty. ClarifAI deduced that the image was related to “travel” and “architecture;” CloudSight detected a pair of white motorboats. However, when light noise was introduced (center image), Watson’s confidence in its bell tower classification dropped to 79%. CloudSight changed its response to “Big Ben” and “Westminster Abbey.” With even more noise added (right image), Watson failed to identify the bell tower at all, while CloudSight hedged its bet on a “white painted concrete infrastructure.” Since we care more about false negatives than false positives (see 3.1.2: Playability for our reasoning), noise presented a greater challenge to our project than the “two giraffes” problem.



Figure 2.9: The Doge’s palace and St. Mark’s bell tower across the grand canal, Venice, Italy; with no noise added (left), some noise added (center), and heavy noise added (right).

2.2.3 Training sets

Training is an important part of IR technology, and allows its instances to specialize in particular fields. Both ClarifAI and Watson IR support custom-trained instances [3] [7]. For example, an instance of the Watson IR service can be trained to become especially good at distinguishing different breeds of dog, but not know anything about other things [7]. However, building up a sufficiently diverse training database can take weeks or even months, posing a significant time investment for intensive projects.

For this project, therefore, the team felt that the amount of time necessary to build an adequate database of images for the test would be out of scope. Through experimentation, we quickly determined that Watson’s default database IR was consistently outperformed by the CloudSight and ClarifAI services, so it was not used.

2.3 Public application programming interfaces

A public API allows a program to utilize another application or service’s computational algorithms through a series of documented requests [17]. This interface lets companies specialize in different services and offer them to individuals or other companies for a fee. The users of the public API service are free to focus on other issues, as they do not need to develop and maintain the API system. With the saved time, however, comes some risk; the company offering the API has complete control of the service and can change or terminate it at will.

2.4 Previous work

Our team’s application is not the first attempt to use IR technology in a game. Other applications, such as *Hwat?!*, *LEGO Fusion*, and *CamFind*, have employed this technology with varying degrees of success. These games’ foci range from looking at a closed set of potential LEGO constructions [6] to providing puzzles whose solutions must be photographed to progress with the game [5].

2.4.1 *Hwat?!*

Hwat?! is a clue-based search game for iOS and Android, available through the Apple App Store and Google Play Store, respectively [5]. The game gives clues, which the user must figure out in order to take a picture of the target object to advance to the next round. Some clues are quite simple (“VROOM VROOM,” see Figure 2.10), but others border on the obscure (“Davinci [sic] Code Chapter 50 Paragraph 2”). Some clues seem unacceptably vague, such as “something that brings you pleasure,” which might be either a toy or chocolate [19].

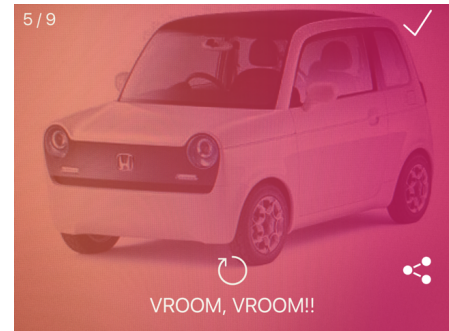


Figure 2.10: Screenshot of the *Hwat?!* application [18].

2.4.2 *LEGO Fusion*

LEGO Fusion is a discontinued series of LEGO sets that let the user transfer their real-life LEGO creations into a virtual LEGO world. The set comes with a few hundred LEGO pieces and a special block called the capture block. Users could construct the façade of a building on top of the capture block using any design they desired. When finished, the accompanying LEGO Fusion app was used to transfer their creation into a virtual LEGO world, where the façade was extruded backwards and transformed into a full 3D building. This extrusion process can be viewed in Figure 2.11, where the user-constructed façade is on the left and the extruded 3D model is shown on the tablet’s screen on the right. Afterwards, the user could play through some of the games included in the app using their own virtual creations [6].



Figure 2.11: *LEGO Fusion* real-life LEGO model (left) with virtual model scanned into tablet (right) [20].

2.4.3 *CamFind*

CamFind is a tech demo for Cloud Sight’s image recognition API. It provides a variety of genuinely useful features which extend the service’s base functionality of simply identifying, categorizing, and searching by images.

After an image is taken, *CamFind* sends it to CloudSight, which returns a wide range of results including include related images, web pages and shopping options. It also provides extended functionality for certain key objects. For example, when presented with a movie poster, *CamFind* will not only identify the movie, but will also bring up movie information, trailers, and showtimes at local theaters.

CamFind includes an interesting social feature that allows users to share search results with friends and family. It also allows users to view other searches in real time. The “Live Stream” and “Popular” features show users’ searches from all over the world. When users find a visual search interesting, they can save the search in their “Favorites” folder for later reference.

2.5 Problems with previous work

There are numerous problems with previous games trying to use IR as a gameplay mechanic. These problems include tricky clues and finicky analysis systems.

Hwat?! does a good job of minimizing processing delay, but has difficulty identifying items consistently, or recognizing objects appearing on brightly-colored backgrounds [19]. Its wild variation in difficulty and lack of a unifying narrative make the game rather dull once its novelty wears out.

LEGO Fusion is often criticized for being picky with analyzing real-world Lego creations. The app requires that the user hold the camera in a certain angle and direction, which can be challenging for young children. Even when the analysis mostly succeeds, users can become frustrated if their creations are not translated into their virtual world with sufficient accuracy.

3 Design

Our project went through many design iterations before reaching its current state. A significant amount of time was spent in our planning and testing phases, as we wanted to ensure that the technologies we employed were the best solutions possible given our needs. The game’s narrative was another major point of revision, often the product of trying to effectively manage the scope of the project.

The decision to evoke a satire of cheaply made free-to-play mobile apps was one of the biggest changes we made, but it proved critical to letting us get away with what was ultimately an aesthetically underwhelming app. This was also an important decision in the sense that it allowed us, all programmers, to focus on the mechanics of the app itself instead of its artistic aspects, and better utilize our limited time to improve the player experience.

3.1 Planning

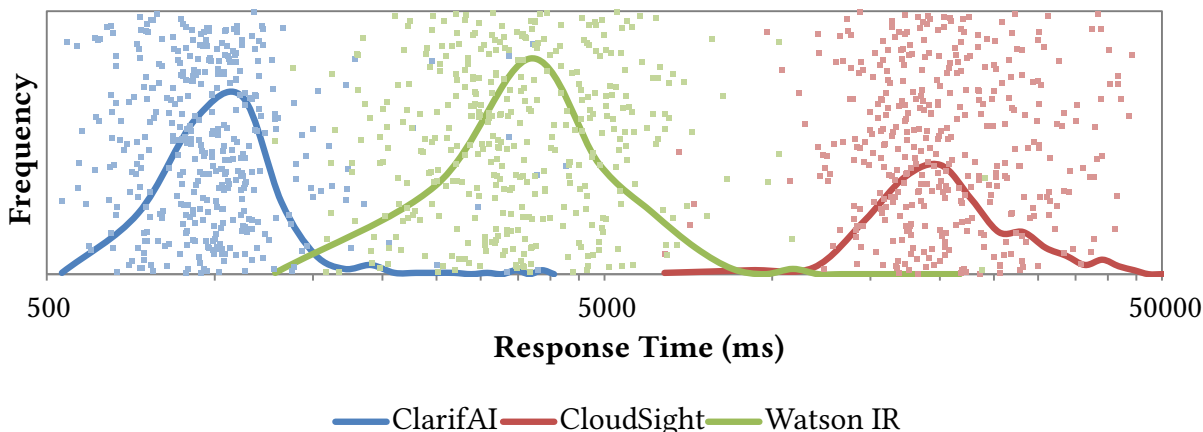
Before we could carry out design for the project, we needed to perform feasibility and performance testing to ensure that the project we were developing could be completed satisfactorily. Our feasibility testing involved repeated testing of several services to determine how quickly and accurately they could identify the types of images we expected to be using.

3.1.1 Exploring public APIs

The team explored three public web APIs: IBM’s Watson IR, ClarifAI and CloudSight. These services were compared by price, ease of use, rate limiting policy, response time and accuracy. It was decided that specificity and the rate of false positives were not priorities for our project, as the team agreed that it was better to have an incorrect answer identified as correct than a correct answer identified as wrong, because the latter issue would cause players to lose engagement. For design purposes, it was assumed that players would not capriciously make random guesses.

To this end, the team carried out testing to compare the rate of false negatives and the response time. A set of 392 images were run through each of the services. The data collected is presented in Figure 3.1.

Figure 3.1: Distribution of Response Times for the services.



Combined with the other data the team collected from the services’ respective websites, such as price and rate limiting (see Figure 3.2), the team decided to use the ClarifAI and CloudSight APIs in parallel, as ClarifAI was significantly faster and CloudSight significantly more accurate than the other services.

Figure 3.2: Comparison of the three services considered.

	Watson [7]	ClarifAI [3]	CloudSight [8]
Price	\$10.00 + 0.2¢/call	0.12¢/call	4¢/call
Rate Limiting	none	none	15/min.

3.1.2 Playability

In addition to its technical feasibility, we wanted to ensure that our project would be fun and interesting to play. To gauge its playability, we created a paper prototype of how we thought the game would look and behave. This prototype included a phone and a set of different panels (see figure 3.3) that could be swapped on and off of the phone. These panels would switch out based on the buttons that were pressed. We took this prototype and informally tested it within our team at the Worcester Art Museum (WAM). One person would read the rules of the game, another person would switch out the panels, and the third person would play through the game. Through this procedure, we were able to estimate that it would take an average of two and a half minutes for a round to be completed. By approximating the additional time required for basic app navigation, and taking into account that we wanted our game to be completed within thirty to forty-five minutes, we determined that the game should include ten rounds of the scavenger hunt.

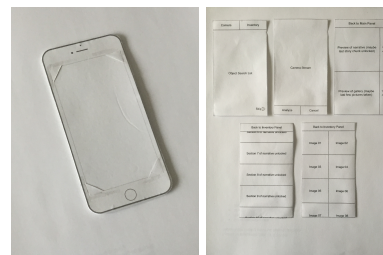


Figure 3.3: Phone paper prototype.

3.2 Game mechanics

The game consists of the main gameplay loop (the scavenger hunt), two secondary mechanics involving a simple form of augmented reality, and a final puzzle activity, together with an assortment of minor one-off mechanics which add support for the game’s background narrative.

3.2.1 Scavenger hunt

It became clear early in the design process that a narrative framework was needed to hook players into the game and encourage them to continue playing more than a few rounds at a time. Flashy dollar signs, confetti raining down the screen, and progressively less underwhelming titles were good, but more was needed to really tie the experience together. In the vein of other games-as-information-gathering-tools such as *Ingress* and *Pokémon Go*, we decided that a scavenger hunt mechanic would complement the core gameplay experience of identifying and capturing elements of images. If we were developing this game to a full release, we would consider using player gathered data in a feedback loop similar to the aforementioned apps, both in a general classification sense and to help augment an image training set as discussed in Section 2.2.3: Training sets, further improving the game.

In the final alpha form, the player is presented with a randomly generated, non-repeating set of three items at a time. Finding any of the three images counts as a success and advances the player to the next round. A success is determined when the IR technology services send back at least one tag that matches the current list of viable classifications. If the player is unable to locate any of the suggested items within a set amount of time, the option to skip that set of items is presented, and a new set is generated.

3.2.2 Augmented reality

In the game, the player learns about a company that is secretly in contact with and receiving symbols from a group of aliens. These symbols have been hidden in museum paintings, and the company has decided to outsource the work of finding them to innocent players who are unaware of the company’s true intentions.

As the game progresses, the app starts to “glitch,” inadvertently showing the player the symbols they are secretly collecting through a semi-augmented reality (AR) mechanic. The mechanic works by overlaying a symbol onto a picture taken by the player. As the player holds their phone, the app uses the phone’s gyroscope as input to produce a parallax “depth” effect. After a few seconds the app then “glitches” out of the semi-AR mode and back into the normal game sequence.

3.2.3 The final puzzle

The game's final challenge is very different from the rest of the experience. Rather than asking the player to search for an item, it requires the player to slide around and superimpose the symbols they collected while playing the game. Neither the meaning of the symbols nor the reason for the inclusion of this mysterious puzzle are conveyed to the player explicitly. Curious players willing to investigate will discover that the symbols can be combined to form an unexplained date in the late 21st century.

One of the difficulties of making a non-explicit puzzle is that its solution must be a sufficiently interesting reveal that the player who goes out of their way to discover and solve it feels sufficiently rewarded for their effort. However, the activity cannot be an integral part of the game. Since its presence is unannounced and never explained, players must be able to feel that the game is complete without solving it, or even being aware of its existence.

Another difficulty we encountered with the final puzzle is that of intuition and instruction. It is considered poor design to introduce additional instruction for single puzzles; for this reason, it was necessary that the puzzle not require instruction for the majority of people who intend to complete it. We decided that sliding symbols around was a sufficiently intuitive puzzle mechanic that it could be understood without introduction, and there is no penalty for failing to reach the solution.

3.2.4 Side mechanics

The game also contains a variety of side mechanics, mostly designed to flesh out the narrative. The most notable of these is the chat panel. Heavily inspired by the indie first-person shooter *Super Hot*, this panel is simply a prerecorded loop of "dialogue" between other players using the app. It is a simple, often humorous way of creating the illusion of a much larger community using the app. This reinforces the false impression sense that the app is actually a generic free-to-play app with a shoehorned social media component. The prerecorded loop is just that, a loop of text bubbles between "users." The player is also able to enter their own text and add it to the "stream" of communication, but since there isn't any real intelligence or player base at the other end, any messages supplied by the player have no effect.

Additional minor components include a parody of an "update screen," which creates the narrative appearance of an emergency, comedic, quick patch when the true nature of the app is accidentally revealed to the player early in the game. Furthermore, a "secure call panel" was developed, which shows up towards the end of the game to create the illusion that the player is overhearing a secret conversation, further driving the narrative. Finally, we have the "rewards panel", which simply shows a mostly artificial representation of the player progressing through the ranks of the game, complete with confetti, trumpets, huge point values, and annoyingly cheerful music.

3.3 Graphic design and user interface

One of the greatest challenges our team faced was the lack of an artist. This meant that any assets and visual design that the team undertook needed to be simple enough to be executed by the team members, whose art skills are decidedly limited.

3.3.1 Parodying free-to-play

The decision was made to parody the so-called “freemium” model for mobile games, which are familiar to smartphone owners and in abundant supply. Our goal was to produce a game mechanic and aesthetic that mimicked these games, allowing us to establish in the player an expectation of how the game would probably function, an expectation which we could then proceed to shatter. This led to the creation of the gopher mascot, Gideon, who was originally intended to dance around on the rewards panel and in other places in the game. Our utter lack of animation skills made this goal unattainable.

In addition to the mascot, several other aspects of the app’s appearance and behavior were inspired by the kitschy presentation of most freemium games. Higher score values increases player motivation [21], a fact which is exploited in many freemium games. Our application takes this to an absurd extreme, awarding points in increments of \$2.5 million.

The update screen and rewards panel (see Section 3.2.4: Side mechanics) are similarly drawn from the design conventions of the freemium model.

3.4 Game narrative

Our game’s narrative framework went through two major phases. The first phase involved a plot in which mysterious symbols have been concealed in paintings by an elite guild of artists, working in secret over the course of centuries. The player has been hired by an unnamed and somewhat sinister company to help uncover the meaning of the symbols.

Eventually the team realized that a more discreet narrative approach might be more intriguing. The second phase of the narrative involved duping the player into believing thinking they were playing an innocent scavenger hunt game, but later uncovering a secret plot wherein the company behind the app is using the symbols collected by the player to assemble a fourth-dimensional portal that will enable an alien invasion of Earth.

3.4.1 Initial attempts

The original backstory for this second approach was fairly grandiose. The idea was that aliens had visited famous painters throughout history and coerced them to encode secret symbols within

their paintings, which the app (which was reverse-engineered from alien technology) is capable of detecting. It is eventually revealed that the aliens are attempting to infiltrate Earth, and the climax of the game centered around a massive invasion event. After finishing the game, it is revealed that this invasion “really occurred,” and that the player is actually living in some form of alien-constructed reality.

This preliminary attempt at a narrative helped us devise a number of fun ideas, such as the augmented reality mechanic, the chat panel and the symbol-sliding puzzle, which eventually found their way into the final game.

3.4.2 Final backstory

The backstory of the final version of our game was similar to the original only in that a group of aliens was attempting to take over the Earth with help from a company called Fourth Kind, Inc. (4KI). However, rather than 4KI explicitly hiring the player, we made the relationship more indirect and hidden. In our revised story, the game we made as part of the project, *The Great Art Quest*, was developed by the fictional company Go 4! Games, a wholly-owned subsidiary of 4KI. The player, then, is inadvertently assisting 4KI in its quest to bring the aliens to Earth.

The hidden images that the player uncovers over the course of the game assist 4KI in its quest, but are never meant to be seen by the player. Fortunately, it appears that the developers at Go 4! Games were in a bit of a hurry, and never had time for proper testing. Glitches in the app eventually allow the player to perceive the dark purpose behind the game they’ve been enjoying.

The final puzzle (see Section 3.2.3: The final puzzle) reveals an arbitrary date in the late 21st century, the significance of which is never explained. Is this when the aliens will finally reach Earth? The world may never know.

3.4.3 Plot outline

The game starts as a cheesy free-to-play game in which the player must find an object in a painting, take a picture of it, and receive coins for a successful find. After a few rounds of the regular game, the app “glitches” and shows the player a hidden symbol in their picture through the augmented reality panel.

In the next round of objects the app pretends to update in order to fix the issue that players are able to access the augmented reality panel. In subsequent rounds the update turns out to be fruitless and the player is viewing a new hidden symbol nearly every round. In later rounds the chat panel becomes more active with “other” players noticing these glitches and commenting on them. Towards the end of the game the app “glitches” in a new way and allows the player to listen in on a call between 4KI and the aliens. In the end the player is directly contacted by 4KI and is informed that there is no hope of stopping the sinister events already set into motion. The

player is finally left with an image gallery full of symbols that they can superimpose over each other to uncover a hidden date.

3.4.4 Completed narrative

The game starts like many other apps, with a splash screen indicating the title and developer of the game. The main game panel is then shown and the mascot of the game, Gideon the Gopher, explains how the game is played. The player is given a set of three objects and needs to find one of them in a painting. The player finds an object, takes a picture of it, and is told by the app whether it is correct or not. If incorrect the player must try again; otherwise, they are rewarded with coins. This scavenger hunt process is referred to as a regular round.

Three more regular rounds are completed before the app “glitches” into the augmented reality panel. Here the player is shown the hidden symbol in the picture they have taken. After a few seconds, an error message appears apologizing for the inconvenience and the player is returned to the game. A regular round follows and the app then “updates” to prevent the player from reaching the augmented reality panel again. A new message appears in the chat panel linking a news headline that Go 4! Games has been connected to 4KI.

Another regular round passes and the update does not work; the player still sees the augmented reality panel. A new chat message asks if anybody else is noticing the “glitches” in the game. The message then becomes censored by the chat admin and is changed to a cheesy comment about how fun the game is.

A seventh regular round is completed and the player experiences a new “glitch”. The player finds themselves in the Secure Communication panel hearing the aliens speaking to someone else. After the next regular round, a new message appears talking about communication interference happening around the world.

The ninth regular round ends and the player listens to the aliens again, but this time there is a human responding to the aliens. The human reassures the aliens that everything is going according to plan. The call ends abruptly after the other parties notice that the player is in on the line and listening.

The tenth and last regular round of the game is completed. This time the player is sent to the Secure Communication panel on purpose, and is informed by an employee of 4KI that they have inadvertently enabled an inevitable invasion of Earth. The player is left to discover the gallery and overlay the symbols they have collected in order to decode the mysterious date.

4 Development

The development of the application was focused on creating a working alpha build of the application that could be tested publicly at the WAM. The alpha build of the application was targeted to be feature-complete, though the team's lack of an artist meant that it was impossible to produce a beta (asset-complete) build of the application. The application was developed using the Unity engine with the C# programming environment, which was chosen over the JavaScript environment for its threading- and sequencing-ready syntax.

4.1 Process

Our development was completed by working on the most essential parts of the application first and then moving on to less essential sections. The most essential feature of the game was to use the mobile camera to take a picture of an object and identify it using public APIs. Less essential components, in descending order, included: narrative panels, rewards panel, and the ending puzzle panel.

4.1.1 Core mechanic

The core mechanic of the game includes three different systems: an object generator, a camera stream system, and a picture analyzer. The object generator and the camera stream system work independently while the picture analyzer relies on both to complete its goal.

The object generator started off as a simple system that choose three random objects from a list. Later on it was extended to make sure that objects would not be repeated until the whole list was exhausted.

The camera stream system helps identify the camera, start the stream, end the stream, digitally zoom the camera, and store a picture from the stream.

The picture analyzer takes a picture and decides if the player has found one of the objects they were supposed to find. It does this by sending a request for the picture to be tagged to ClarifAI and CloudSight. This request is sent out as an HTTP `POST` request to each service with the body including the image. As the app receives results from each service, it compares the tags received with the list of acceptable tags for each object that the player is looking for. If a match is found, then the system proceeds to the next panel in the narrative whether it be the Rewards panel, Augmented Reality panel, or the Secure Communications panel.

4.1.2 Side panels

The development of the side panels was divided among the team members, and was completed in several phases based on a priority system in which panels that connect more closely to the core mechanics had the highest priority, followed by panels of narrative importance, then decorative panels. The first panels to be created, therefore, were the main screen and the camera screen. Then were completed the rewards panel, the update panel, the secure call panel, and the tutorial. The last steps were the chat panel and the image gallery.

4.2 Testing

Testing was critical to our development cycle. To comply with WPI's institutional review board (IRB) requirements, our testing pool consisted entirely of WPI students. We ultimately conducted three playtesting sessions throughout the year, ranging from paper prototypes to get a sense of player strategy, mechanical feasibility, and round duration, to final validation on a fully functional alpha build running on a real phone.

4.2.1 Building a testing plan

We wanted to verify that our core mechanics were fun and intuitive. To do this, we made testing at the WAM a priority. Aside from some simple exit surveys (see Appendix D: Testing survey) the premise was to bring people to the museum, give them the app, and observe what happened. Many bugs were discovered in this way, especially in early rounds of testing. One big area of concern early in the development cycle was making sure that players were successful at finding the items for which they were looking. Beyond completion time and success rate, other metrics were gathered through the surveys, such as player experience, difficulty, aesthetic impressions, and effect on museum engagement.

4.2.2 Execution

Testing occurred in three rounds. Each round was conducted identically, except that the first two rounds were thirty minutes in length and the final was an hour. Participants were brought into the WAM, and were given a phone with the app installed. No further instruction was given except that they should play the game to the best of their ability. After playing the game for the set amount of time, the test was stopped, and players were given a survey (see Appendix D: Testing survey). Testing was done confidentially and anonymously, as specified by the Informed Consent form included in the IRB application (see Appendix E: IRB protocols). A total of four people participated in the first testing round, five in the second, and two in the third.

5 Evaluation

The following is a brief evaluation of the successes and failures of our project. We also discuss our motivational goals as described in Section 1: Introduction of the paper, as well as changes that could have been made to streamline the process of production and improve the end result of the project.

5.1 Testing results

We performed three rounds of testing to evaluate our game. The first two rounds consisted of 30-minute sessions used to identify operational and user interface issues. The third and final round involved 60-minute sessions designed to give players time to finish the entire game and provide us with a more summative evaluation of the app.

5.1.1 Summary of Round 1

The first round of testing involved four participants, two playing on the Android platform and the other two playing on iOS. From this round, we learned that there were significant operational bugs that were missed during development testing. The most serious problem was frequent app crashes due to mismanagement of memory allocation. Other, less significant issues were also noted, such as the phone's camera being flipped horizontally on the Android platform, as well as a lack of visual cues to the image analysis going on in the background.

Figure 5.1: Graph of the effect of the app on museum engagement according to the playtesting survey results for round 1.

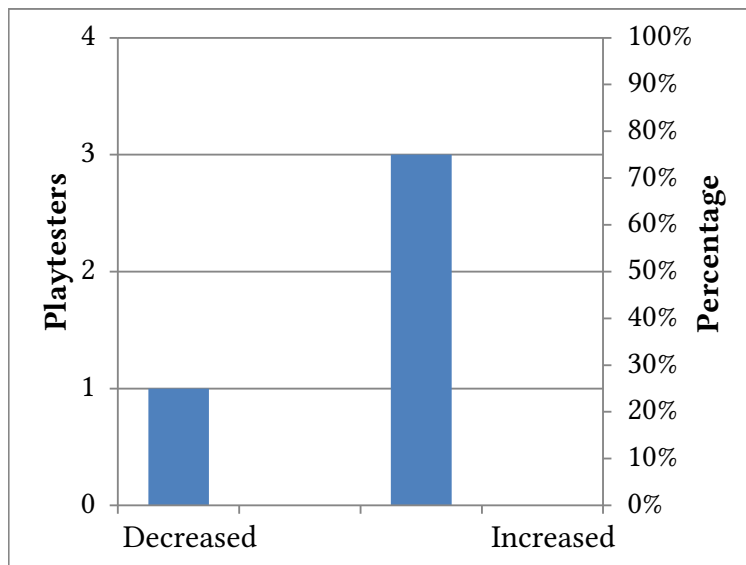
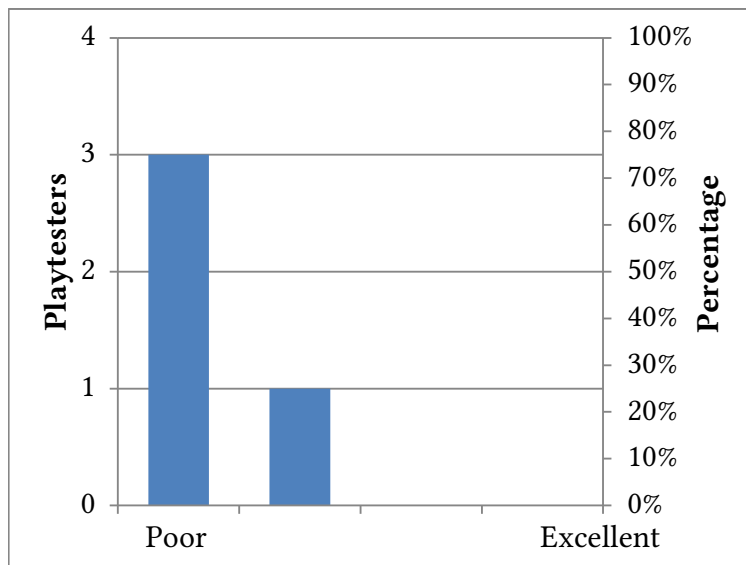


Figure 5.2: Graph of the success of the app to identify images according to the playtesting survey results for round 1.



Of the four playtesters in this round, one stated that the app decreased their engagement with the museum and the other three stated that it moderately increased their engagement (see figure 5.1), and opinions on the accuracy of the application were unanimously negative (see figure 5.2).

5.1.2 Summary of Round 2

The second round of testing involved five participants. While app crashes were less frequent, they were caused by sound cue issues rather than memory allocation. There was also a misunderstanding between the development team that resulted in three of the participants only being able to submit images to ClarifAI and not CloudSight. From this misunderstanding we learned that CloudSight's longer but more accurate response helped lower player frustration. This also tended to expedite gameplay, because players spent less time retaking pictures.

Of the five players in this round, only one stated that the application decreased their engagement with the museum, with three citing a moderate increase and the final declaring a large increase in engagement (see figure 5.3). Opinions on the accuracy of the application in determining correctness of images was split, with three indicating good performance and two indicating poor performance (see figure 5.4).

Figure 5.3: Graph of the effect of the app on museum engagement according to the playtesting survey results for round 2.

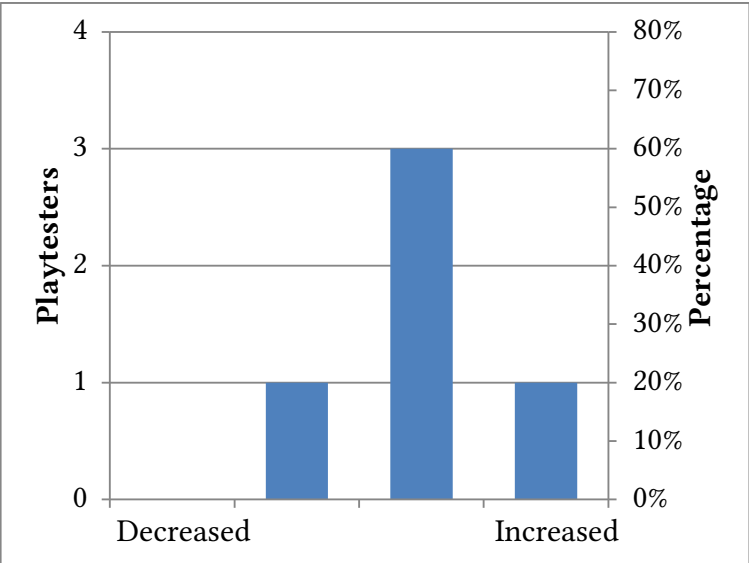
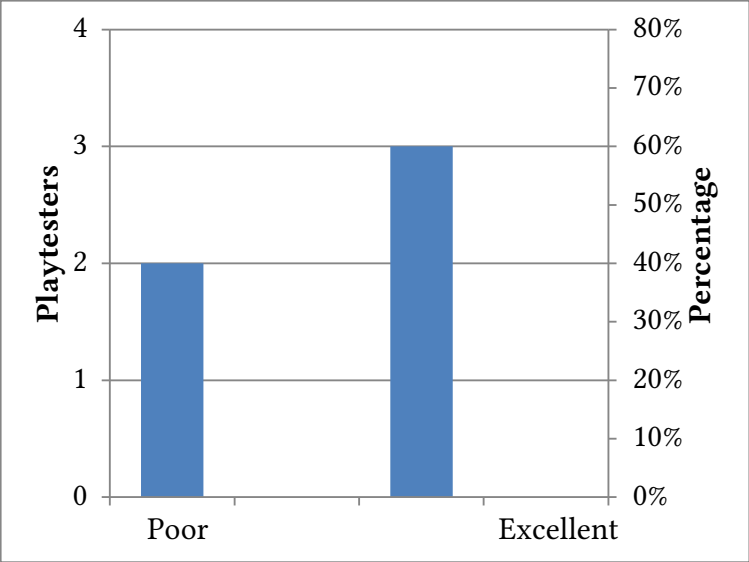


Figure 5.4: Graph of the success of the app to identify images according to the playtesting survey results for round 2.



5.1.3 Summary of Round 3

For the final round of testing we only received two participants. Each participant was able to finish the game well within the one hour session they were assigned to. Also this time the game did not crash at all, and both ClarifAI and CloudSight were available to the player. Instructions still could have been improved, but overall the game was an enjoyable experience.

The results of this round were middling with regards to perceived accuracy, with one player scoring the accuracy as moderate and the other as good. There was, however, general agreement on the effect on museum engagement: both participants stated that the app moderately increased their engagement with the museum.

5.2 Capability of IR technology for interactive media

Though the testing we performed is too limited to be statistically significant, it does suggest that IR technology can be successfully integrated into games as a core gameplay mechanic. Thanks to the reasonable accuracy of the IR services used by our app, our playtesters were able to play through the majority of the game even in the short thirty minute sessions of rounds 1 and 2.

5.3 Possible future improvements

There are certainly aspects of the project we could have improved upon. The most obvious opportunity for improvement is the audiovisual presentation of the app. Because we did not have any artists on our team, the visuals in our app could not hope to recreate the quality of professional titles. Our unpolished art also diminished the impact of the fictional “glitches” in the game.

Another, more technical area for improvement is the potential to dynamically adjust the scavenger hunt list based on a museum’s location. At present, the app features a relatively generic list of objects, but there are strategies we could deploy for customizing the object list to better suit each museum. One way to accomplish this would be to cull objects from the list that have a low or zero hit rate, and to dynamically add objects to the list based on tangential image recognition data (i.e. other objects recognized in transmitted images), or add them manually with the aid of museum staff or online catalogs of featured art.

Finally, depending how frequently it’s used on the same paintings, false negative results could be used to train the service to recognize images that it currently struggles to classify. For example, WAM contains a painting featuring a very prominent knife that is rarely if ever recognized by the services, despite being frequently selected by participants. Specific subset machine learning (a feature built in to many IR solutions) could be used to train the services to detect such objects.

Works cited

- [1] (Apr. 5, 2017). Deep learning, Wikimedia Foundation, Inc., [Online]. Available: https://en.wikipedia.org/wiki/Deep_learning (visited on 04/05/2017).
- [2] (2017). Machine vision company, Machine Vision Company, [Online]. Available: <http://www.machinevc.com/> (visited on 02/14/2017).
- [3] (2016). Artificial intelligence with a vision, ClarifAI, [Online]. Available: <https://www.clarifai.com/> (visited on 02/08/2017).
- [4] (2017). TapTapSEE, TapTapSEE, [Online]. Available: <http://taptapseeapp.com/> (visited on 02/14/2017).
- [5] J. Bland and R. Havrylyuk. (2016). *Hwat?! Resentient*, [Online]. Available: <http://www.hwat.io> (visited on 01/26/2017).
- [6] N. Lee, “*Lego Fusion* lets you build virtual playgrounds with real-world bricks,” *Engadget*, Jun. 2014. [Online]. Available: <https://www.engadget.com/2014/06/19/lego-fusion/>.
- [7] (2017). Watson, IBM, [Online]. Available: <https://developer.ibm.com/watson/> (visited on 02/01/2017).
- [8] (2017). Visual cognition, CloudSight, [Online]. Available: <http://cloudsight.ai/> (visited on 02/08/2017).
- [9] (2017). Unity, Unity, [Online]. Available: <https://unity3d.com/> (visited on 02/14/2017).
- [10] B. G. Silber and T. Triplett, “A decade of arts engagement,” *National Endowment for the Arts*, Jan. 2015. [Online]. Available: <http://www.arts.gov> (visited on 02/14/2017).
- [11] B. Folkens, *CloudSight Full Logo*. CloudSight, 2015. [Online]. Available: <http://cloudsight.ai> (visited on 02/08/2017).
- [12] J. Dove, “CamFind launches CloudSight API to advance visual search,” *The Next Web*, Feb. 2015. [Online]. Available: <https://thenextweb.com> (visited on 02/08/2017).
- [13] (Nov. 21, 2014). CloudSight privacy policy, CloudSight Inc., [Online]. Available: https://cloudsight.ai/privacy_policy (visited on 04/12/2017).
- [14] J. Markoff, “Computer wins on ‘Jeopardy!’: Trivial, it’s not,” *The New York Times*, Feb. 2011. [Online]. Available: <http://www.nytimes.com/> (visited on 02/01/2017).
- [15] (Jul. 23, 2015). Why Google’s Deep Dream AI hallucinates in dog faces, Co. Design, Inc, [Online]. Available: <https://www.fastcodesign.com/3048941/why-googles-deep-dream-ai-hallucinates-in-dog-faces> (visited on 04/12/2017).
- [16] P. Vasuki, C. Bhavana, S. M. M. Roomi, and E. L. Deebikaa, “Automatic noise identification in images using moments and neural network,” in *2012 International Conference on Machine Vision and Image Processing (MVIP)*, IEEE, Dec. 2012, pp. 61–64. doi: 10.1109/MVIP.2012.6428761. (visited on 02/08/2017).
- [17] M. Rouse. (Jan. 2015). What is open API? [Online]. Available: <http://searchcloudapplications.techtarget.com/definition/open-API> (visited on 02/08/2017).

- [18] R. Havrylyuk. (Oct. 19, 2016). *Hwat?!* screenshot, TouchArcade.com, [Online]. Available: <http://forums.toucharcade.com/showthread.php?p=3930583> (visited on 02/06/2017).
- [19] J. Bland. (Oct. 2016). *Hwat?!* long-form demo video. Resentient channel, YouTube, [Online]. Available: <https://www.youtube.com/watch?v=djZU86URNco> (visited on 02/06/2017).
- [20] “CellPhone”. (Jun. 21, 2014). *Lego Fusion* merges physical brick building electronic kids toys digital play, TouchArcade.com, [Online]. Available: <http://topcellsmartphone.com/lego-fusion-merges-physical-brick-building-electronic-kids-toys-digital-play/> (visited on 04/04/2017).
- [21] W. Hao and S. Chuen-Tsai, “Game reward systems: Gaming experiences and social meanings,” in *DiGRA #3911 - Proceedings of the 2011 DiGRA International Conference: Think Design Play*, DiGRA/Utrecht School of the Arts, Jan. 2011. [Online]. Available: <http://www.digra.org> (visited on 02/13/2017).

Appendix A: Game design document

(Document begins on next page)

The Great Art Quest

Game Design Document

Designers:

Thomas J. Meehan
Yahel Nachum
Andrew L. Ray

Advisors:

Brian J. Moriarty
Charles Rich

I. Elevator Pitch

The Great Art Quest is a game available for iOS and Android devices that uses image recognition (IR) application programming interfaces (APIs) to build a unique experience in which players take photographs of paintings in art museums and uncover a sinister plot.

II. Audience

The main target audience of *The Great Art Quest* is 13–21-year-old smartphone users that likely would not go to a museum otherwise. An older audience that would visit a museum already is a secondary target audience.

III. Narrative Summary

The player begins by finding objects at the request of the game's mascot, an animated gopher named Gideon. Once the player succeeds in taking a picture of the desired object, the game rewards the player with confetti, a fanfare, 2.5 million points, and possibly even a new rank. Each one of these repetitions of getting a list of objects, finding them, and getting a reward will be called a *round*.

After several rounds, the game appears to glitch out, and a symbol is shown overlaying the photo that the player took to get the item. The player is notified that an error has occurred and developers are being notified to fix the problem... immediately. After one more round, an “update” to the app will be installed. However, this update does not fix the issue, and the glitch recurs.

By this point, the player has received hints through the chat panel that Go 4! Games, the purported developer of *The Great Art Quest* has been linked to a suspicious company called Fourth Kind, Inc. (4KI), and a couple rounds later, the player enters the “secure call” panel, where they intercept a communiqué between 4KI and what sounds to be an alien.

After another round, the player gets contacted by 4KI directly. They tell the player that 4KI is aware of the intercepted call. The player is informed that it is too late for them to save the world, but as a consolation, the player will get double points in the next round. After the final round, a final puzzle is given, which may be solved by looking in the image gallery at the mysterious glyphs that showed up over the course of the game. These combine to form a date, the meaning of which is left intentionally vague.

IV. Characters

- Player
- Gideon the gopher
- 4KI employee
- Alien

V. Mechanics

One of the game’s mechanics is giving the player a set of objects and the player needing to go in search for one of them in the art museum. A set of objects is used in order to allow alternative options to the player in case one of the objects is not present in their art museum.

Another mechanic involves the player taking pictures with their mobile camera of the objects they are looking for. These pictures will be used to determine if the player has found the correct object. The computer will determine if the object is correct by using online visual recognition services. The service will bring back a list of items that it believes are in the picture. If the computer determines that they have not found the correct object then the player will have to keep searching the art museum for the correct object. If the computer determines that the player has found the correct object they will unlock the next section of the game.

The final puzzle uses a different mechanic, of moving images around on the screen so that they can overlap to form a date.

VI. Game Flow

Upon opening the app for the first time the player would receive a thirty-second- to one-minute-long narrative introduction to the basic premise of the game. After that would follow a guided introduction of the basic mechanics. The player would be given a choice of three different items to find in paintings. The player would then walk around the museum to find one of those three items. Once the player finds one he or she can take a picture of it. When the camera is active, the game would outline a section of the frame (likely a square) in which the desired object would be contained (and ideally fill) that area.

The area within that square would be the area analyzed by the IR service. The picture would be analyzed by an AI while some sort of animation played in order to cover up the computation. If the painting does not contain the object the player is told to try again. If the painting contains one of the objects the player is notified as such and is given a new set of three objects. Each painting is assigned a point value based on difficulty, the percentage match of certainty given by the AI (thus incentivizing good content), and additional factors. If they player cannot find any of the three objects a “skip” button with a cooldown of about five minutes would allow them to skip objects.

This pattern continues until a set number of points is achieved, after which a major narrative milestone is unlocked, potentially combined with level ups and player progression. Higher levels could require more points to finish. Narrative “artifacts” could be viewed in a separate inventory menu after they are unlocked.

Appendix B: Narrative document

(Document begins on next page)

The Great Art Quest

Narrative Document

Designers:

Thomas J. Meehan

Yahel Nachum

Andrew L. Ray

Advisors:

Brian J. Moriarty

Charles Rich

Overview

The game consists of 11 narrative chunks. Each chunk describes what will happen in that section of the game in terms of the story being revealed.

Chunk 0

Once the phone finishes loading the app the player is greeted with the main menu panel. The Gopher mascot is seen jumping up and down with joy. The Gopher thanks the player for trying out the beta. The Gopher tells the player that it is glad to see them. The gopher then gives the player a brief overview of how the game works.

The Gopher points to each of the items on the screen and tells the player that they need to find at least one of these items in a painting. Afterwards the Gopher points to the camera button and tells the player to click that button once they have found one of the items.

Once the player clicks the button the Gopher tells the player to pinch and zoom the camera to align the item in the screen. The Gopher also points to the Analyze button and tells the player to click the button to indicate that they have found the item. If the analysis fails the Gopher tells the player to try again. If the analysis succeeds the Gopher congratulates the player and shows them their new stats.

This cycle of getting a list of objects, finding them in the museum, taking a picture, and being rewarded will be referred to as a regular round in subsequent sections.

Chunk 1

A regular round occurs omitting the tutorial bits.

Chunk 2

Another regular round is completed.

Chunk 3

The player goes through a regular round.

After the analysis succeeds the game starts to “glitch”. The “glitch” includes showing the player the augmented reality panel. In this panel the player can view the glyph they found in the painting. After a few seconds the game pauses. A message appears telling the player that “the app seems to be experiencing issues and that the developers are currently working on fixing it... immediately”. The player then dismisses the message and the game switches back to the main menu panel.

Chunk 4

The player goes through a regular round.

After the picture is taken, the player is taken to an updating page and is told that the app is experiencing a mandatory update. The player is told that none of their other data will be affected, parodying the Windows 10 update. Elevator music plays as the player waits.

Once the update is completed the chat panel includes a message with a news headline saying that “Go 4! Games linked to mysterious ‘4th Kind Inc’ says FBI”.

Chunk 5

The player goes through a regular round and the glyph glitch occurs again. A new message appears in the chat panel reading, “Hey, has anyone else seen a weird pic pop up when they win?” As the player is reading this message, it is changed by the chat admin, first to “Censored by ADMIN”, and then to “This game is the best!”

Chunk 6

The player goes through a regular round and the glyph glitch occurs again. After the player receives coins from their successful analysis the game glitches into the secure communication panel. The player starts to hear aliens talking on the other end of the connection. The game abruptly glitches back into the main menu panel.

Chunk 7

The player goes through a regular round and the glyph glitch occurs again. During this round the chat message quotes headlines such as “Strange communication interferences plague entire globe.”

Chunk 8

The player goes through a regular round and the glyph glitch occurs again. Shortly after the player completes this round, they start to hear a conversation between 4KI and the aliens. 4KI reassures the aliens that the building of the machine is going according to plan and that it will be completed on schedule.

Chunk 9

The player goes through a regular round and the glyph glitch occurs again.

After the player receives coins from their successful analysis the game switches to the secure communication panel. A worker at 4KI greets the player and tells them that 4KI knows that the player was listening in on their call. The 4KI worker fills in the player on what’s happening

and explains that they don't have any time to stop it. Then call ends sending the player to the main panel.

Chunk 10

The player is not allowed to proceed regular round cycle of the game. They are encouraged to look around the rest of the game and possibly stumble upon the image gallery panel. Inside the image gallery are the glyphs seen earlier in the game. The player then arranges the glyphs in such a way that it reveals a secret message.

Appendix C: UI design

(Document begins on next page)

The Great Art Quest

UI Design Document

Designers:

Thomas J. Meehan
Yahel Nachum
Andrew L. Ray

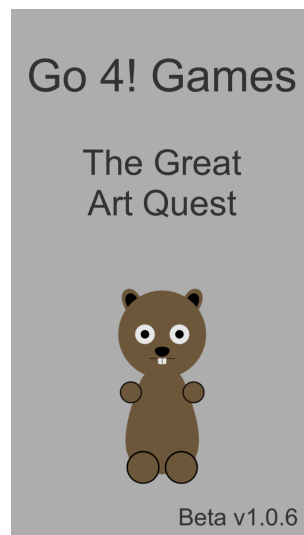
Advisors:

Brian J. Moriarty
Charles Rich

I. Introduction

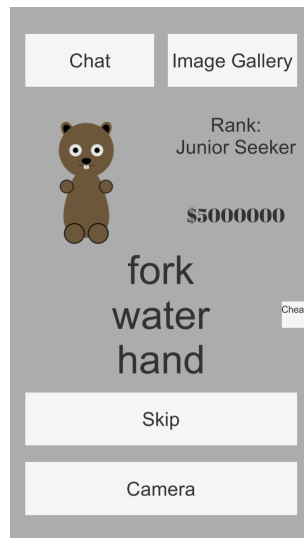
This document includes a description of all the panels used in the game

II. Splash Screen Panel



This panel pops up whenever the app loads up. It is just like any other splash screen and shows the name of the game, the company that made it, an identifying feature of the game (the gopher), and it plays a small little start-up song.

III. Main Panel



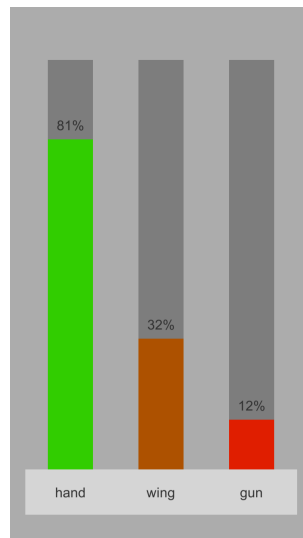
This panel is the main hub of the game. From here the player can go to the Camera Stream Panel, Chat Panel, and Image Gallery Panel. They can also view their rank, number of coins received, and the objects they are trying to find. If they get stuck on a difficult set of objects they can hit the skip button after a certain amount of time to receive a new set of objects to find.

IV. Camera Stream Panel



On this panel the player can digitally pinch and zoom the camera, take a picture of an object they found, and send it to be analyzed. If the player changes their mind they can also cancel and return to the Main Panel. Lastly, at the top of the panel there is a smaller object list to remind the player of what they are supposed to be searching for.

V. Analysis Panel



The analysis panel is shown once the player has sent an image to be analyzed. It randomly animates the bars to give the player the feeling that it is deciding whether or not they have found

one of the objects. If the player succeeds it sends them to Rewards Congrats Panel, otherwise it sends them back to the Camera Stream Panel.

VI. Rewards Panels

There are two sections to the rewards panel: 1. congratulating the player, and 2. showing them their new statistics.

VI.1 Rewards Congrats Panel



On this panel the player is congratulated for finding one of the objects. A stream of confetti is thrown over the panel and a sound effect is played saying “Yay!” At the bottom a next button sends the player to the Rewards Stats Panel.

VI.2 Rewards Stats Panel



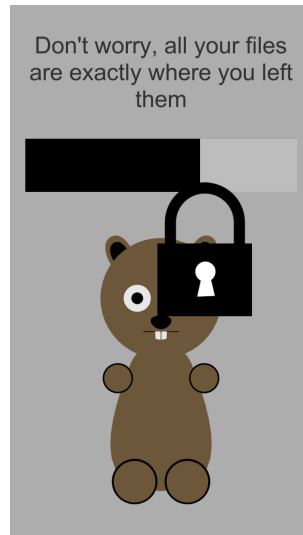
This panel displays the player's new statistics including their new rank and the amount of coins they received. The done button sends the player to the next panel in the narrative sequence.

VII. Augmented Reality Panel



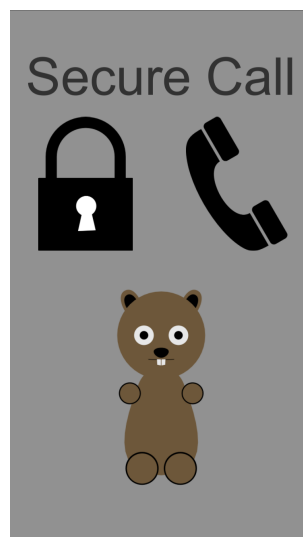
The Augmented Reality Panel shows the player a superimposed glyph over the picture they sent to be analyzed. As the player tilts and rotates their phone it induces a parallax effect on the screen using the glyph and the background image.

VIII. Update Panel



The update panel shows a loading bar with different text in the header relaying information to the player. The header text is parodying the Windows 10 update screen.

IX. Secure Communication Panel



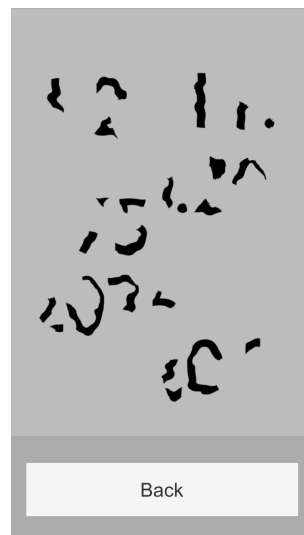
This panel mostly consists of different communication calls that the player listens in on.

X. Chat Panel



The Chat Panel simulates other players in the game that can indirectly interact with the player by presenting them with new narrative information.

XI. Image Gallery Panel



As the game goes on the Image Gallery Panel fills up with new glyphs that the player uncovers. These glyphs can be rearranged to reveal a secret message.

Appendix D: Testing survey

Note: Asterisk (*) denotes a question that was used to evaluate the project's motivational goals. (1-4) indicates that participants were asked to give a rating from one to four.

The first section of the survey was to be filled out by the administrator

1. Operating System
 - (a) Android
 - (b) iOS
2. Testing Round
 - (a) Round 1
 - (b) Round 2
 - (c) Round 3
3. Did the player complete the game within the allotted time?
 - (a) Yes
 - (b) No

The remainder of the survey was to be completed by the participant

4. What are the game's rules and objectives? How did you discover them?
5. How easy was it for you to understand and operate the game? (1-4)
6. Did you encounter any specific difficulties playing the game? If so, briefly describe them.
7. In general, how easy was it for you to locate the specified objects? (1-4)
8. Were any objects particularly difficult to locate? If so, which ones?
9. How would you rate the game's ability to accurately identify objects? (1-4) *
10. How would you rate the level of challenge presented by the game? (1-4)
11. Did any aspect of the game particularly surprise or impress you?
12. How would you describe the game to someone who has never played it?
13. How do you think the game affected your engagement with the museum? (1-4) *
14. Do you have any general comments or questions?

Appendix E: IRB protocols

(Document begins on next page)

Informed Consent Agreement for Participation in a Research Study

Investigator: Brian Moriarty

Contact Information: bmoriarty@wpi.edu

Title of Research Study: MQP MBJ 1700 AI Entertainment Applications

Sponsor: N/A

Introduction

You are being asked to participate in a research study. Before you agree, however, you must be fully informed about the purpose of the study, the procedures to be followed, and any benefits, risks or discomfort that you may experience as a result of your participation. This form presents information about the study so that you may make a fully informed decision regarding your participation.

Purpose of the study: The purpose of this study is to obtain feedback on the above-named Major Qualifying Project in order to facilitate design improvements and find/address operational bugs.

Procedures to be followed: You will be asked to play a game lasting 15-30 minutes. After completing the game, you will be asked to complete a brief, anonymous survey describing your subjective experience.

Risks to study participants: There are no reasonably foreseeable risks associated with this research study.

Benefits to research participants and others: You will have an opportunity to enjoy and comment on a new art game under active development. Your feedback will help improve the game experience for future players, and its published MQP report will lead to an improved understanding of the game design/development process.

Record keeping and confidentiality: The survey you complete as part of this research will contain no personal information. It will be secured in a password-protected database, and deleted at the conclusion of the project.

Records of your participation in this study will be held confidential so far as permitted by law. However, the study investigators, the sponsor or its designee and, under certain circumstances, the Worcester Polytechnic Institute Institutional Review Board (WPI IRB) will be able to inspect and have access to confidential data that identify you by name. Any publication or presentation of the data will not identify you.

Compensation or treatment in the event of injury: There is no reasonably foreseeable risk of injury associated with this research study. Nevertheless, you do not give up any of your legal rights by signing this statement.

For more information about this research or about the rights of research participants, or in case of research-related injury, contact the Investigator listed at the top of this consent form. You may also contact WPI's IRB Chair (Professor Kent Rissmiller, Tel. 508-831-5019, Email: kjr@wpi.edu) and the University Compliance Officer (Jon Bartelson, Tel. 508-831-5725, Email: jonb@wpi.edu).

Your participation in this research is voluntary. Your refusal to participate will not result in any penalty to you or any loss of benefits to which you may otherwise be entitled. You may decide to stop participating in the research at any time without penalty or loss of other benefits. The project investigators retain the right to cancel or postpone the experimental procedures at any time they see fit.

By signing below, you acknowledge that you have been informed about and consent to be a participant in the study described above. Make sure that your questions are answered to your satisfaction before signing. You are entitled to retain a copy of this consent agreement.

Study Participant Signature

Date: _____

Study Participant Name (Please print)

_____ Date: _____

Addendum for WPI IRB Application Form

Title: MBJ-1700 AI Entertainment Applications

Purpose of study

To obtain playtest feedback in order to locate/address operational bugs, and to identify opportunities for design improvement.

Study protocol

Participants are provided a handheld device on which to play the game. Investigators observe participants during play. Afterward, participants are asked to fill out a short survey to characterize their subjective experience.

Hazardous materials/special diets

No hazardous materials or special diets are involved in this study.

Opening briefing for testers

“Hello, and thank you for volunteering to test our game. Before we begin, could you please read and sign this Informed Consent form? [Tester signs IC form.] Thank you. When your session is complete, we will ask you to complete a brief survey about your play experience. At no point during your play session, or in the survey after, will any sort of personal and/or identifying information about you be recorded. Please begin playing when you feel ready.”

Questions for post-test survey

1. What are the game's rules and objectives? How did you discover them?
2. How easy was it for you to understand and operate the game?
 - (1-4 scale, 1 = Difficult, 4 = Easy)
3. Did you encounter any specific difficulties playing the game? If so, briefly describe.
4. In general, how easy was it for you to locate the specified objects?
 - (1-4 scale, 1 = Difficult, 4 = Easy)
5. Were any objects particularly difficult to locate? If so, which one(s)?
6. How would you rate the game's ability to accurately identify objects?
 - (1-4 scale, 1 = Poor, 4 = Excellent)
7. How would you rate the level of challenge presented by the game?
 - (1-4 scale, 1 = Difficult, 4 = Easy)
8. Did any aspect of the game particularly surprise or impress you?
9. How would you describe the game to someone who has never played it?
10. Any general comments or questions about the game?