

I Don't Believe My Eyes! Geometric Sketch Recognition for a Computer Art Tutorial

D. Cummmings & F. Vides & T. Hammond¹

¹Department of Computer Science & Engineering
Texas A&M University, College Station, TX, USA

Abstract

Drawing is a common form of communication and a means of artistic expression. Many of us believe that the ability to draw accurate representations of objects is a skill that either comes naturally or is the result of hours of study or practice or both. As a result many people become intimidated when confronted with the task of drawing. Many books and websites have been developed to teach people step-by-step skills to draw various objects, but they lack the live feedback of a human examiner. We designed EyeSeeYou, a sketch recognition system that teaches users to draw eyes using a simple drawing technique. The system automatically evaluates the freehand drawn sketch of an eye at various stages during creation. We conducted frequent evaluations of the system in order to take an iterative development approach based on user feedback. Our system balances the flexibility of free-hand drawing with step-by-step instructions and realtime assessment. It also provides rigorous feedback to create a constructive learning environment to aid the user in improving her drawing. This paper describes the implementation details of the sketch recognition system. A similar implementation method could be used to provide sketching tutorials for a wide number of images.

Categories and Subject Descriptors (according to ACM CCS): Computer Applications [J.5]: ARTS AND HUMANITIES—Fine Arts; Software [D.2.2]: SOFTWARE ENGINEERING—Design Tools and TechniquesUser Interfaces

1. Introduction

The presence of art and artistic expression can serve as an influence to who we are as scientists and researchers [FK95]. Art can be seen as a form of communication and a means of expressing one's interpretation of the world. We all aren't born with the skill or have the time to learn how to accurately interpret and reproduce what we see around us through drawing. While the task of learning how to draw perfect reproductions of all what we see may be improbable on a large scale and within a small time frame, it may be possible on a much smaller scale and with the aid of technology. Numerous books and websites have been written to explain in detail how to draw various objects through step-by-step drawing tutorials [Mas04, Smi07]. As sketch recognition algorithms and technologies become more sophisticated, it is now possible for a computer to watch a human draw, use sketch recognition techniques to identify what the human is trying to draw, apply difference classification algorithms to

determine the accuracy of what they are drawing, and then use this information to provide realtime interactive feedback. We have decided to explore this theory by focusing on how to draw one of humans' most expressive features: the eye.

Identifying basic structures and shapes within life and recreating those shapes through various methods is the basis of most technical drawing instruction [Par79, WP88]. A proper representation of these structures falls under the drawing approach taught by most art teachers and summarized by the phrase "draw what you see, not what you know" [Edw99, Mar08]. The meaning of this direction is to overcome the urge to draw the shape that the mind determines is correct and to instead draw the shape that is actually seen.

A novice artist can reference an extensive collection of videos and how-to literature focused on improving artistic skills. Many manuals have been written focusing on a similar approach of breaking down drawing techniques into steps

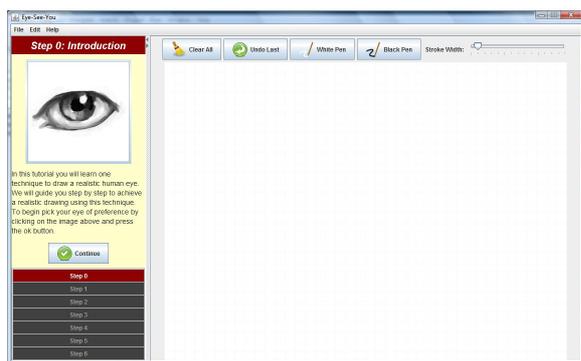


Figure 1: *EyeSeeYou interface*



Figure 2: *Sample eye drawn with the EyeSeeYou system.*

that can be self-taught [DeR96, Par03]. However, as is often the case with how-to drawing methods, the final result is only as good as your ability to accurately recreate the progressive images.

Without direct observation and guidance from an instructor, a novice artists may find it difficult to realize when they're making a mistake; when they do realize, there is usually no guide to show them how to get back on track. In order to address this problem we took a simple technique for drawing an eye and developed a system that would provide feedback on how well the user follows this technique; this system is called EyeSeeYou (Figure 1).

EyeSeeYou is a pen-based system that uses sketch recognition to provide helpful feedback to users based on their drawing. As no two freehand drawn sketches are alike, accurately recognizing hand-drawn sketches is difficult. By walking the user through a step-by-step drawing process similar to those found in paper-based drawing tutorials, the system is able to remove ambiguity in terms of the user's intention, and provide informative and useful feedback. The system allows users to sketch their drawing as they would with paper, enabling the user to learn a particular technique defined in a series of steps, while making sure the user follows the instructions at each step, so they can later reproduce the same technique in their drawings.

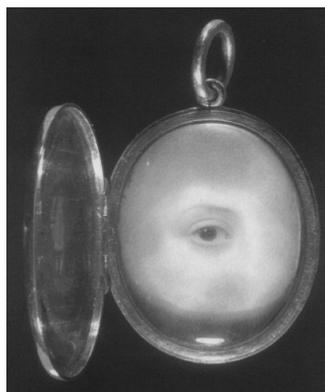


Figure 3: *Eye portraits encased in small jewelry, such as pins and rings, were popular in the 18th century, often used to mourn loved ones. Cosway, *The Eye of Mrs. Fitzherbert*, ca. 1786. (Photographic Survey, Courtauld Institute of Art).*

The EyeSeeYou application uses a geometric-based approach to recognition; it uses a low-level recognizer called PaleoSketch [PH08] to identify primitive shapes such as lines, arcs, ellipses, spirals, etc. The system uses a set of guidelines and constraints to verify that the user draws elements of the eye proportionally and in correct relation to each other. If the procedure is followed correctly, then the result should be a structurally correct eye that can then be enhanced similar to the one shown in Figure 2.

The main challenge in the creation of the system was to find a balance of constraints that would allow the system to deliver constructive feedback while at the same time preventing user frustration from over-correction. In order to accomplish this task we attempted to create a systematic approach to drawing the eye by breaking it down into simple steps that can be completed fairly quickly and with reinforcing feedback that both helps the user to identify where his or her drawing could use improvement. We also wanted to reinforce progress made toward the production of a realistic eye with positive feedback.

An initial challenge was to determine how strict the constraints for the drawing task had to be. If they were too loose the final result may not be sufficiently realistic, but if they were too strict they may have an adverse affect on the user experience as it may become too difficult to advance at each step.

An additional challenge involved trying to guide the artist while still allowing a level of sketching freedom so as not to impede the creative process. Our intention was not to force user to create exact reproductions of the example image that is given, but instead, we want to provide them with the techniques to create their own image of an eye that is proportionally accurate. The images that we obtained after evaluating our software show that although the eyes drawn by each of

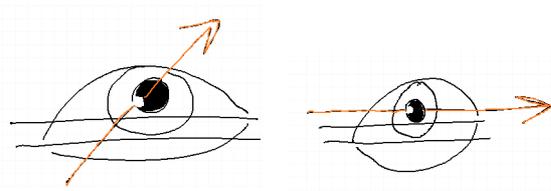


Figure 4: *Two very different, but correctly shaded eyes in EyeSeeYou*

the participants may be similar, the level of detail and individual features are still unique.

2. Previous Work

The significance of the eye in communication can be seen through popular art forms of the 18th century [Fer75]. A method of portraiture emerged that consisted of small miniature paintings of the eye [Gro06]. These unusual portraits were often mounted on pins, brooches, rings and other keepsake items and exchanged between friends, family and lovers as a sign of affection. In some cases, eye portraits of the deceased (called mourning jewelry) were kept in remembrance of loved ones. An example is shown in Figure 3. Although short-lived, the popularity of this type of portrait gives us insight into the importance the eye has in conveying specific meaning relating to both the subject (the owner of the eye) and the viewer. It's not the still image of the eye, but more importantly, the gaze that holds meaning and context for the viewer. In the case of eye portraits, the image of an eye would hold less meaning to someone who did not recognize it. As a result, each portrait had a very private and intimate affect on their owners. While this method of portraiture is no longer prevalent, we chose the eye as our subject of interest because of the simplicity of the drawing method in conjunction with the overall impact of such a small image.

The use of technology as a means of assisting users to draw has been explored in various drafting programs and modeling applications. However, more attention has been focused on systems that facilitate image beautification [WSP05, BBS09, BJ96], rather than sketch recognition and learning [ODG04, TH09]. In addition, these programs usually focus on drawing objects other than those found in the life drawing areas such as faces or human figures.

ShadowDraw is one of few applications that guides the user to create free-hand life drawings [LZC11]. ShadowDraw generates a shadow image in real-time to guide the user while she sketches a drawing. The shadow is created from a database of similar images and allows the user to trace over the shadow to create a new image.

Tsang et al. used a similar concept to develop a suggestive interface that aids users in creating 3D wireframe models [TBSR04]. The program provides 2D images of the mod-

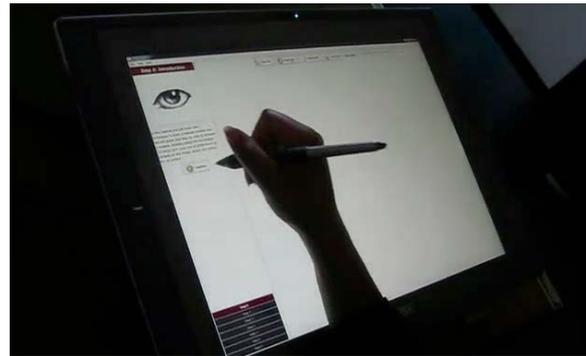


Figure 5: *Wacom monitor with pen input*

els as a guide to get users started. As they begin to add strokes, the system prompts them with suggestions of 3D parts that can be added to the model. These parts are augmented to the existing model thereby facilitating the drawing process. This technique is very powerful and allows creative drawing. However the recognition of the sketch being drawn is very limited or in existent which inhibits the ability of giving meaningful feedback to the user.

While systems like these help the user by augmenting their drawings, they do not provide feedback that would help the user create better drawings on their own. In order to provide this extra level of interaction, sketch recognition was used to create predictive and suggestive interfaces that guide the user to construct a particular object. iCanDraw? is a system that uses sketch and face recognition in order to evaluate the user's input [DPH10]. Face recognition is used to generate a model of a human face from a photograph; this model is then used as a basis of comparison for the sketch recognition algorithms. iCanDraw? provides feedback to the user when their input does not accurately correspond to the generated model and the system guides them to make certain modifications in order to get back on track. iCanDraw relies on template matching for most of its recognition. This is perhaps the major difference with EyeSeeYou, where no explicit template is used. Instead, a set of constraints are defined for each step allowing the user to draw any eye as long as it follows the specified constraints. The added benefit of using constraints over template matching is that an original drawing is not required as a baseline for comparison, this frees the user from being confined to reproduce a specific image. In this sense, our approach conforms more to a style of drawing instruction that encourages free-form drawing while adhering to general rules of aesthetics. EyeSeeYou does not force the user to reproduce the sample image given, but evaluates various images correctly as long as they follow the guidelines given. For example, the lighting reference lines can be in any direction as long as the reflections and shading on the iris are in agreement 4. The goal is

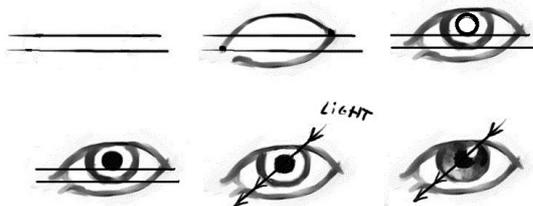


Figure 6: 6-steps to draw an eye

to teach the concepts behind drawing a perceptually correct eye with or without a model image present.

The use of constraints pertains to the field of geometric recognition. Geometric recognition has been explored and researched in various distinct domains. In this kind of recognition there is usually a bottom-up approach and after pre-processing, there is a low level recognizer that can identify primitive shapes such as lines, circles or arcs. On top of that recognition there is a higher level recognizer that can use a set of constraints to determine if the basic shapes and the relationship between them compose a more complex shape. This approach has been used successfully in domains such as military Course of Action [JH10], or circuit diagrams [Alv04] [GKSS05]. In all cases a combination of primitive shapes under a set of known constraints results in the production of higher level shapes that comply to certain standards, yet allowing free sketching.

3. IMPLEMENTATION

The EyeSeeYou system was designed and developed with a pen-based interface as primary input. The user can draw sketches on the screen using a Wacom monitor and an electronic pen which records the user's strokes on a digital canvas as shown in Figure 5. These strokes are understood by the computer as a series of two-dimensional points sampled in time. We can then use these data to feed a sketch recognizer that will identify primitive shapes. These shapes will be used as further input to a sub-system that checks that the drawn shapes comply with the expected drawing step within a certain threshold of confidence.

3.1. 6-step Lesson

EyeSeeYou was implemented using a 6-step procedure in order to teach the user the proper technique for drawing an eye [Smi07]. These steps are based on a method that relies on the size, position and orientation of the shapes drawn by the user relative to reference lines (Figure 6). The six steps are to 1) draw reference lines, 2) draw the outline of the eye,

3) draw the pupil and iris, 4) fill the pupil, 5) draw the direction of a light source, 6) add reflections and shading. The instructional steps provided to the user are as follows:

Step 0: In this tutorial you will learn one technique to draw a realistic human eye. We will guide you step by step to achieve a realistic drawing using this technique. To begin pick your eye of preference by clicking on the image above and press the ok button.

Step 1: The first thing we need is a reference to draw our eye. One common mistake in drawing eyes is to assume they are completely symmetric. In reality the inner corner (the one closer to the nose) is lower than the outer corner as can be seen in the picture. Our first step is to draw 2 parallel reference lines to base the measurements of our drawings. Use the canvas to draw 2 lines parallel and close to the center.

Step 2: Now that you have a reference you can draw the outline of the eye. Using 2 smooth continuous strokes, try to form the outline as shown in the picture. This may take a little practice. Notice that the lines are curved, but do not form a perfect circle. Also notice that there is almost twice as much space above the top reference line.

Step 3: Draw the pupil. These are 2 concentric circles inside of the outline where the inner circle is filled. Make sure the outer lines are round as shown in the image

Step 4: Fill the pupil. The inner circle should be black so go ahead and fill it in.

Step 5: Draw the reference line for the shading. This is an arrow that goes in the direction of the light source as shown in the picture. Make sure the arrow intersects the pupil as shown in the image. This will be useful to guide yourself in the next step.

Step 6: With the white pen you can draw a white speck where the light hits the eye. In the opposite end of the light the iris will be lighter, go ahead and draw the shadows as shown in the figure.

3.2. Interface

We designed a simple and intuitive interface for the user to follow the drawing tutorial. At the left of the screen the system provides the user with pictures and instructions on how to complete each step (including how to draw the guidelines) and gives a brief explanation on why that step is recommended. After the user draws the image depicted in the window according to the instructions, she can click the Continue button to proceed to the next step. After the system analyzes the image, if it determines that feedback is necessary to help the user improve the image, the system will display a dialog box with detailed feedback so the user can attempt to correct the image by undoing the last strokes or clearing the screen to try again. Once the drawing is correct the user can proceed to the next step. At any moment the user can use the navigation buttons on the left to move between steps. However, because this is a step by step procedure, the user is not allowed to skip steps, and only previous steps are enabled.

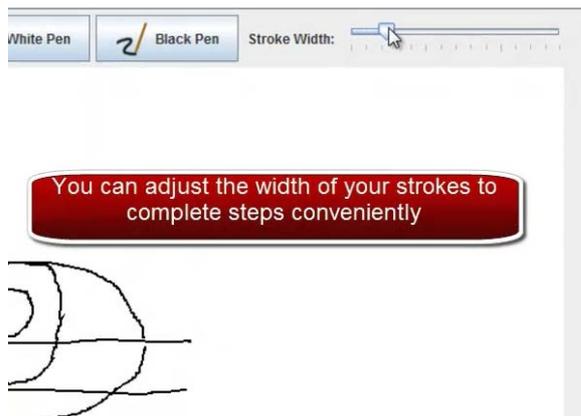


Figure 7: Larger width strokes should be used for coloring in various parts of the eye.

The interface includes a tool palette (Figure 7) that allows users to modify the width and color of their strokes and to enable shading and lighting features in order to make the eye look more realistic.

3.3. Low level recognition

We use a low level recognition system to extract the beautified version of shapes drawn, which we can then use to calculate various constraints. The model that we use to determine the expected shapes and constraints amongst them at each step is based on the dimensional features of the human eye that have been extracted by artists and anthropologists so that the eye will seem natural [KK01, Par79]. We translate these features into a set of primitive shapes and the relationships between them. EyeSeeYou recognizes the primitive shapes that should be present at each step of the drawing and evaluates their position and characteristics relative to reference lines (Figure 6). These lines also help guide the user as to where to accurately place the elements of the eye. The goal is that later the user can follow this technique using regular pen and paper.

After capturing the raw data given by the pen, EyeSeeYou uses a low-level recognizer called PaleoSketch [PH08] to identify the primitive shapes in the sketch. PaleoSketch integrates several techniques such as corner finding and geometric perception to do a series of pre-recognitions over the supported shapes. It then uses a novel ranking algorithm to determine which of these shapes has a better fit. Although the current version of PaleoSketch supports more than 10 basic shapes; we are mostly relying on the recognition of lines, polylines, curves, arcs and circles. PaleoSketch itself has a reported accuracy of more than 98%. Every time there is a pen-up event we identify the new stroke and feed it to the high level recognition subsystem.



Figure 8: EyeSeeYou showing corrective feedback

3.4. High level recognition

In the high level recognition subsystem, we try to ensure that at each state in the tutorial, the user has drawn progressive images according to the model we have established. When all the steps in the procedure are followed correctly, the result is a structurally accurate eye; this is an eye that follows regular proportions.

We maintain the primitive shapes recognized by PaleoSketch in a data structure so that each time the user clicks continue, the system evaluates the sketch on screen with what is expected in each step. Each step contains a model that is represented as a set of required primitives and the constraints on the relationships of these primitives. Along with these constraints comes the confidence used to determine if the sketch complies or not. We do not want a constraint that is too tight, since it will become frustrating to the user to use the system as it would be very difficult to replicate each step. But we also do not want to relax the constraints too much as the final result may be a poorly drawn eye. We therefore chose not to have absolute thresholds, but instead dynamic ones that will depend on the context. For example, to determine if the two reference lines are spaced too far apart we set a maximum distance relative to the average size of the two lines, instead of having a fixed distance. Although each step will use the same concept, we use fine tuning that is specific to each step.

In Step 1, the initial reference lines need to be parallel and the spacing between them in relation to their length must be proportional to the one in the picture, otherwise the system provides corrective feedback (Figure 8). To verify this, we first use the low-level recognizer to determine if the strokes are lines (and not curves, points, etc). Then we compare the distance between the left most point of the top line and the left most point of the bottom line with the distance between the right most point of the top line and the right most point of the bottom line. If the distances vary by more than 25% then it implies that the lines are either not parallel, or vary significantly in length. If the user's line do not meet the constraints, the system will display a message such as

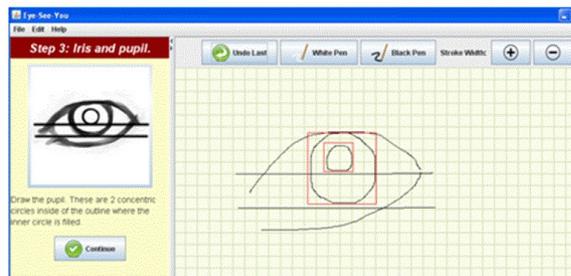


Figure 9: Comparison of pupil and iris

"Make sure your lines aren't spaced too far apart", or "Make sure your lines are parallel", whichever is appropriate or both if necessary.

In Step 2, the user is asked to draw the outer line of the eye. The tutorial [Smi07] states that the eye is supposed to slant downward towards the nose. In this case we present the user with a left eye, so EyeSeeYou compares the endpoints of the lines drawn by the user to the reference lines drawn in Step 1. The lines that make up the outline of the eye must meet the following criteria in order to be evaluated as correct by the system:

1. Both the upper and lower outlines should be arcs (primitive shape recognized by PaleoSketch).
2. The left most endpoint of the upper outline of the eye must begin below the top reference line and above the bottom reference line. The Y value of the left most endpoint of the outline is compared to the average Y values of the reference line strokes. If the Y value of the stroke point is higher, then this shows that the eye outline stroke begins above the bottom reference line. The following comparisons are performed in a similar manner.
3. The rightmost endpoint of the upper outline must end above both reference lines.
4. The leftmost endpoint of the bottom outline must end below both reference lines.
5. The rightmost endpoint of the bottom outline must meet the rightmost endpoint of the upper outline.

If the strokes are not recognized as arcs, the system displays a message letting the user know that her outlines need to be a smooth curve in order to represent the curve of the lids above and below the eye.

In Step 3, the user is asked to draw the pupil and the iris of the eye. EyeSeeYou verifies that these two strokes resemble circles and provides feedback to the user if they don't. EyeSeeYou also uses the intersection of the bounding boxes to verify that one of these circles is completely contained inside the other (Figure 9).

The position of the bounding boxes are then used to verify that the pupil and the iris are in proper position relative to the reference lines. The bottom Y values of each bounding

box is compared to the average Y values of the reference line strokes to ensure that the Y values of the bounding boxes are greater and that the pupil and the iris lie above the top and bottom reference line respectively.

In Step 4, the user is asked to fill in the pupil. EyeSeeYou provides a slider to increase the stroke width of the pencil in order to facilitate this step. EyeSeeYou verifies that the stroke drawn to fill in the pupil lies relatively inside the pupil outline stroke by comparing the overlap of the bounding boxes. If the percentage of the bounding box of the pupil stroke that is contained within the bounding box of the pupil stroke is 95% or greater, then the system accepts the stroke.

In Step 5, the user is shown how to draw an arrow that represents a light source directed on the eye. In the example image, the light source is shown as coming from the upper right corner, however EyeSeeYou does not apply a constraint on the direction. EyeSeeYou looks for and recognizes a primitive arrow shape using low-level recognition and proceeds to the final step if found.

In Step 6, the user is asked to draw a reflection on the pupil of the eye in relation to the reference line drawn in the previous step. EyeSeeYou provides a feature to change the color of the pencil from black to white so that the user can draw the white reflection over previous strokes.

Once the user has successfully completed Step 6, she has an accurate basic outline of an eye that should be fairly similar to the final example shown. She can then use the drawing tools provided in EyeSeeYou to add embellishments to the image in order to make the eye look more realistic.

During the development process, we conducted 3 preliminary studies with groups of 2 - 5 people. The purpose of these studies was to test the software for errors and to obtain iterative feedback on the usability and design of the system. This feedback, discussed in detail in the following section, was instrumental in helping to create a balance of instruction and correction that was conducive to learning.

4. EVALUATION

In the artistic world, evaluation is particularly challenging since a measure of aesthetics cannot be easily quantified. In addition, there are various points of view that argue in favor of early and continuous user studies using quantitative data; while others state that usability evaluation can sometimes have negative impacts on the project if considered as the only evaluation tool [Ols07]. We still believe that user studies are very significant in this domain as they allow us to gain insight on the system's usability and effectiveness, but it is also important to count on an additional measure besides our own appreciation of the collected sketches. Instead we relied on expert assessment (in this case an artist and instructor) to evaluate the system and the results produced.

We recruited a group of ten participants with varying levels of self-assessed artistic skill and experience working with

sketch-based applications. We used a Wacom Cintiq 21UX tablet as the input device on a standard desktop computer (Intel Core2 2.6GHz, 4GB Ram, running Windows 7). We first showed each participant an image of an eye and asked them to use the Wacom to draw the eye free-hand based on their current artistic knowledge. After they finished drawing their eye, we asked them to open the EyeSeeYou program and proceed through the lesson. We recorded the initial free-hand image as well as the final image produced after completing the lesson for comparison. We recorded the amount of time each user took to complete the lesson as well as the number of attempts made to proceed through each step. We tried to determine how effective the instructions were and/or if the users read them before proceeding with the drawing task. We also tried to observe the level of difficulty users had with completing the tasks at each step. Some of the participants had little or no experience using a pen-based application and were unsure about the amount of pressure to apply and/or at which angle to hold the pen. As a result, sometimes users made (what they thought were) mistakes while trying to complete the steps and would often redraw the image multiple times before submitting it for evaluation. We allowed the participants to use the monitor at various tilt levels for ease of use and determined that most of the participants felt comfortable drawing on the Wacom tablet at an angle of about 45 degrees in relation to the table. At the end of the study we gave the participants a short survey to obtain additional feedback about their experience.

5. Results

5.1. User Experience

The user experience in this case is a very important feedback for us, as the main goal of this program is to be user-oriented. This part of the survey tried to determine if the user felt comfortable with using the software. Some improvements were made based upon their comments. For example, if users expressed frustration or difficulty with a task, we often relaxed the constraints up to a point that would not sacrifice the integrity of the image. Also, we made updates to the verbiage of the lesson directions if participants were confused by the text.

In the survey, we asked them to qualify their overall feeling about using a digital pen, beginning the lesson, the directions of each step, the feedback that the program offered, the ease of drawing and advancing to the next step and their overall experience upon completing the drawing. Each of these aspects was ranked by the user in a scale from 1 to 5 and we conveyed the results in (Figure 10). We found that the users were comfortable using the pen, even those who had no prior experience with pen-input devices. They showed no trouble beginning the lesson and that the instructions and feedback were meaningful most of the time. However many felt that the constraints were still too tight so advancing through each step and completing the lesson was

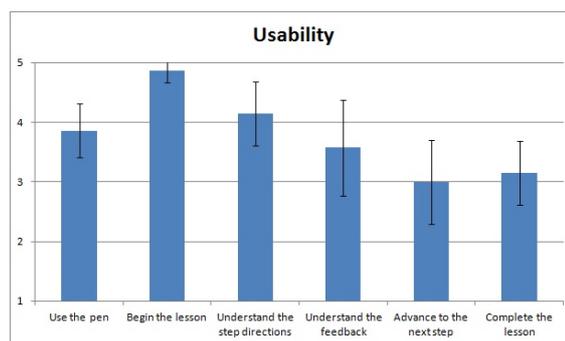


Figure 10: Usability measure in the system

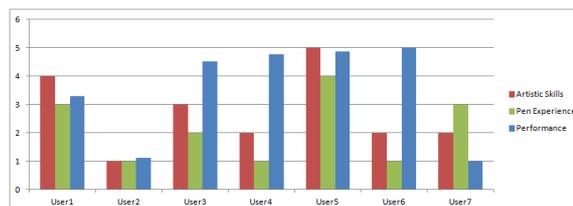


Figure 11: User-defined skill level compared to performance

not very easy. It is important to note that most of the users that had this feeling also reported that the presented drawing technique did not coincide with their drawing style.

5.2. User Performance

As we mentioned previously, the task of evaluating a drawing is very subjective; user performance in terms of artistic skill is difficult to quantify. However, there are some aspects that can give us insight into the ability of the user to follow instructions and successfully complete the steps in the lesson. One performance measure is the time taken by users to go through the process. On average the users took 12 minutes to complete the process, however one user did it in less than 6 minutes. Another important metric was how many attempts it took for the users to complete each of the steps in the process (identified by the number of clicks on the Continue button). Based on these two metrics, we graded the user on a scale from 1 to 5 using simple normalization $perf = 5 * (1 - \frac{u-min}{max-min})$, where u is the number of attempts that a particular user needed to complete the task and max and min represent the maximum and minimum attempts of the entire test group. We then then averaged the time and the attempts; the results are summarized in Figure ???. Table 1 shows how many attempts it took on average for all the users to complete each step. We found that steps 2 and 3, which consisted of drawing the outline of the eyes and then the pupil and iris, were particularly difficult. This is understandable considering that this is where most cognitive

Step	Avg. attempts
Step1	3.1
Step2	3.8
Step3	6.9
Step4	1.6
Step5	3.0
Step6	1.1

Table 1: Average attempts per step

issues seem to occur when using pen and paper as well. People tend to draw an oval shape instead of circles, or they draw the pupils and iris out of place as can be seen in the initial free-hand drawn sketches.

5.3. Resulting Images

Unlike the work presented by other researchers, our approach to recognition gives more flexibility in terms of drawing. We verify that the user's image conforms to a model of constraints in order to check the correctness in the structure of the eye rather than a template matching against an existing bitmap or sketch.

Since our main objective is to develop a system that teaches users how to draw a realistic eye, we also needed some measure of effectiveness related to the correctness of the sketch. We thought it best to obtain qualitative input from someone with both professional and teaching experience in the arts. As part of our testing effort, we had an art professor evaluate the system as well as the resulting images from our user study. We showed the professor before and after images of novice participants in the user study in random order. We then asked him to verbally critique the images and rank them in order of increasing aesthetics. The art professor ranked 75% of the images created using EyeSeeYou in the top 50% of the ranking order. During the critique, the professor commented that although some of the eyes appeared to have issues with scaling and orientation, the ones he ranked the highest appeared to show an understanding of the characteristics of the eye such as the location of the iris, angle of the eye tilt, etc.

Figure 12 shows a comparison between the sketches drawn by some users prior to using the system on the left and the version created using EyeSeeYou on the right. We can see marked differences between the images. The beauty of each eye is somewhat subjective, but according to the artists' evaluation, the images on the right tend to match the overall shape of a human eye more accurately.

6. DISCUSSION

The first part of our evaluation focused on the user experience. Although EyeSeeYou supports any pen input device, we wanted to use accessible yet appropriate hardware. The

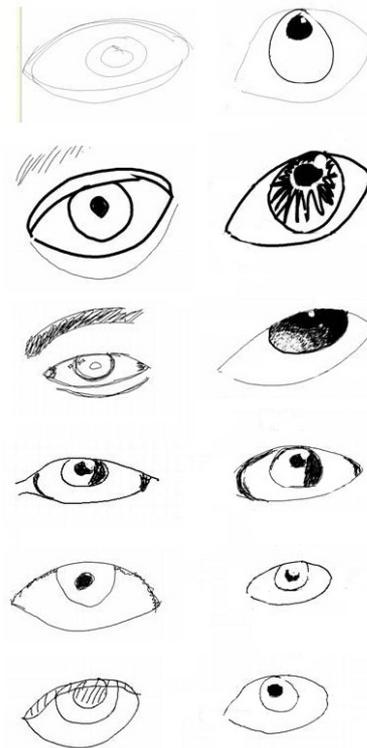


Figure 12: User results, before and after

screens we chose have been on the market long enough for them to be robust and convenient to use. Although digital sketching is not the same as writing with a pencil on a piece of paper, the input interface was very well accepted, particularly with novice users. However, some of the more skilled artists felt more constrained compared to when they use real pen and paper, as the latter allowed them to have deeper control in shading and responded naturally to tilting, pressure and over-tracing. Advanced artists also found it uncomfortable to be directed to use a particular technique since they may have been taught different techniques for drawing the human eye in the past. As a result, we discovered that experienced artists produced images that did not accurately resemble the sample image as much as those of novice artists. Advanced artists tended to feel hindered by the constraints used to evaluate their drawing since they were already familiar with such aesthetic rules.

As for the interface itself, earlier versions of the system caused complaints that the recognition was too restrictive. This is what motivated our choice of using perceptual constraints. Through observation we were able to gain insight on what feedback and guidance methods were most beneficial to the user. Even though participants were instructed to read the instructions carefully before proceeding with the task, some admitted that they skipped the reading and in-

stead simply tried to make their drawing match the sample image shown at each step. Since EyeSeeYou is a drawing program, it makes sense that users would rely on visual cues as a guide for their task instead of text instructions. However, this method quickly falls short considering that a common mistake of a new artist is to draw what she knows instead of what she sees. For example, novice artists often tend to draw eyes as iconic circles, since that represents what they are in real life, although in reality they can only see a small sliver of the eye compared to its whole. In a few of these cases, the system did not allow users to proceed because their resulting image was not proportional. However, there were a few instances where the drawn image would fit the criteria and let the user proceed, however, constraints of a subsequent step would be harder to meet due to lack of attention to these details. Although some steps were nontrivial to users, overcoming the usual cognitive issues was not much of a problem once the feedback received was carefully followed.

The second part of our evaluation focused on user performance. The results show that when using the system, there is very little correlation between skill level and performance (i.e. the skilled artists did not significantly outperform). Perhaps not surprisingly, EyeSeeYou tended to even out the results; it made novice drawings better and expert results worse. Therefore, we believe that our system is most effective for novice artists.

Overall, based on a majority of the comments submitted, users felt that the system was fairly easy to use. The results of our evaluation suggest that the EyeSeeYou system might help some novice artists to learn a new drawing technique. When all steps are completed correctly, the resulting outline represents an image structure that is fairly close to the final image. EyeSeeYou does not intend to be an isolated program, but rather a first step in a much broader impact consisting of a set of tools that uses sketch recognition to teach young artists how to draw a variety of images.

7. Future Work

This prototype served us as a proof of concept that can guide the design of a more complete system, one that includes lessons on drawing all the features of the face. Utilizing similar instruction and recognition methods, we believe we can implement the process to draw elements such as the nose and lips (Figure 13).

Enhancements to the way the feedback is shown to the user would contribute to this effort. The system can be modified to include visual markers for identifying the problem areas within a sketch. It could also provide suggested modifications similar to the Teddy or the 3D drawing interface presented in the related work section. Also there could be error levels or categories, as some minor constraints are desired but not necessarily mandatory. In the current implementation all the errors are stoppers, you can not continue until

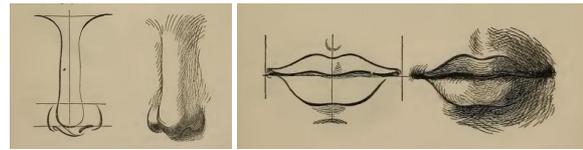


Figure 13: *Nose and Lips drawings with reference lines [UH07]*

they are solved. In a future version of the system a warning type of error might be desired.

Although it was not included in the basic tutorial, another feature would be to add the recognition of proper shading techniques. Accurate shading can prevent the eye image from looking flat and EyeSeeYou would be able to utilize the light source reference line created in step 5 in order to verify the accuracy of the component shadows. Implementing shading may require adjusting the grey scale and width of the stroke according to pressure information of the pen when available.

We could also allow the user to adjust parameters that will modify the tightness of the constraints so novice artists do not get frustrated by over correction. This would also allow users to progressively train themselves to follow a technique until it becomes natural, or they could chose to adjust the parameters for a more challenging drawing experience.

8. Conclusion

EyeSeeYou is a pen-based system that uses sketch recognition methods paired with a simple eye drawing technique in order to guide a person to create an accurate drawing of the human eye. While this prototype system was focused on drawing the eye, we believe the idea can be extended to tutorials for various types of drawings. Other facial features such as the nose and mouth have been characterized in similar step-by-step drawing techniques. With additional enhancements, this system can provide detailed feedback that will further improve the aesthetics of the final result. EyeSeeYou serves as a proof of concept for a system that has the potential to be expanded into a robust learning tool that can be used to teach novice artists how to draw a realistic-looking images.

9. Acknowledgments

The authors would like to thank members of the Sketch Recognition Lab for their contribution to this project.

References

- [Alv04] ALVARADO C.: Sketchread: a multi-domain sketch recognition engine. In *In UIST 2004 ACM symposium on User interface software and technology (2004)* (2004), pp. 23–32. 4

- [BBS09] BAE S.-H., BALAKRISHNAN R., SINGH K.: Everybodylovesketch: 3d sketching for a broader audience. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology* (New York, NY, USA, 2009), UIST '09, ACM, pp. 59–68. URL: <http://doi.acm.org/10.1145/1622176.1622189>, doi:<http://doi.acm.org/10.1145/1622176.1622189>. 3
- [BJ96] BHAVNANI S. K., JOHN B. E.: Exploring the unrealized potential of computer-aided drafting. In *Proceedings of the SIGCHI conference on Human factors in computing systems: common ground* (New York, NY, USA, 1996), CHI '96, ACM, pp. 332–339. URL: <http://doi.acm.org/10.1145/238386.238538>, doi:<http://doi.acm.org/10.1145/238386.238538>. 3
- [DeR96] DERREYNA R.: *How to Draw What You See. 35th Ed.* Ed. Watson-Guptill Publications, 1996. 2
- [DPH10] DIXON D., PRASAD M., HAMMOND T.: ican-draw: using sketch recognition and corrective feedback to assist a user in drawing human faces. In *Proceedings of the 28th international conference on Human factors in computing systems* (New York, NY, USA, 2010), CHI '10, ACM, pp. 897–906. URL: <http://doi.acm.org/10.1145/1753326.1753459>, doi:<http://doi.acm.org/10.1145/1753326.1753459>. 3
- [Edw99] EDWARDS B.: *The New Drawing on the Right Side of the Brain.* Putnam, 1999. 1
- [Fer75] FERRIS W.: Vision in afro-american folk art: The sculpture of james thomas. *The Journal of American Folklore* 88, 348 (1975), pp. 115–131. 3
- [FK95] FINLEY S., KNOWLES K.: Researcher as artist/artist as researcher. *Qualitative Inquiry*, 1, 1 (1995), 110–142. 1
- [GKSS05] GENNARI L., KARA L. B., STAHOVICH T. F., SHIMADA K.: Combining geometry and domain knowledge to interpret hand-drawn diagrams. *Computers and Graphics* 29, 4 (2005), 547 – 562. doi:[10.1016/j.cag.2005.05.007](https://doi.org/10.1016/j.cag.2005.05.007). 4
- [Gro06] GROOTENBOER H.: Treasuring the gaze: Eye miniature portraits and the intimacy of vision. *The Art Bulletin* 88, 3 (2006), 496–507. URL: <http://www.jstor.org/stable/25067264>. 3
- [JH10] JOHNSTON J., HAMMOND T.: Computing confidence values for geometric constraints for use in sketch recognition. In *Proceedings of the Seventh Sketch-Based Interfaces and Modeling Symposium* (Aire-la-Ville, Switzerland, Switzerland, 2010), SBIM '10, Eurographics Association, pp. 71–78. URL: <http://dl.acm.org/citation.cfm?id=1923363.1923376>. 4
- [KK01] KOBAYASH H., KOHSHIMA S.: Unique morphology of the human eye and its adaptive meaning: comparative studies on external morphology of the primate eye. *Journal of Human Evolution* 40, 5 (2001), 419 – 435. URL: <http://www.sciencedirect.com/science/article/B6WJS-45819GD-1F/2/21a0ccce658e44c1b0dd4e50bbff4429>, doi:DOI: [10.1006/jhev.2001.0468](https://doi.org/10.1006/jhev.2001.0468). 5
- [LZC11] LEE Y. J., ZITNICK L. C., COHEN M. F.: Shadowdraw: real-time user guidance for freehand drawing. In *Proceedings of ACM SIGGRAPH 2011* (2011), SIGGRAPH '11. 3
- [Mar08] MARTIN N.: Assessing portrait drawings created by children and adolescents with autism spectrum disorder. *Art Therapy: Journal of the American Art Therapy Association* 25, 1 (2008), pp. 15–23. 1
- [Mas04] MASIELLO R.: *Ralph Masiello's Bug Drawing Book.* Charlesbridge Pub Inc, 2004. 1
- [ODG04] OH Y., DO E., GROSS M.: Critiquing freehand sketching - a computational tool for design evaluation. *Visual and Spatial Reasoning in Desing (VR)* 22, 1 (2004), pp. 105–120. 3
- [Ols07] OLSEN JR. D. R.: Evaluating user interface systems research. In *Proceedings of the 20th annual ACM symposium on User interface software and technology* (New York, NY, USA, 2007), UIST '07, ACM, pp. 251–258. URL: <http://doi.acm.org/10.1145/1294211.1294256>, doi:<http://doi.acm.org/10.1145/1294211.1294256>. 6
- [Par79] PARISER D. A.: Two methods of teaching drawing skills. *Studies in Art Education* 20, 3 (1979), pp. 30–42. URL: <http://www.jstor.org/stable/1320170>. 1, 5
- [Par03] PARKS C.: *Secrets to Drawing Realistic Faces.* North Light Books, 2003. 2
- [PH08] PAULSON B., HAMMOND T.: Paleosketch: accurate primitive sketch recognition and beautification. In *Proceedings of the 13th international conference on Intelligent user interfaces* (New York, NY, USA, 2008), IUI '08, ACM, pp. 1–10. URL: <http://doi.acm.org/10.1145/1378773.1378775>, doi:<http://doi.acm.org/10.1145/1378773.1378775>. 2, 5
- [Smi07] SMILSHKALNE TEST L.: Eye tutorial. URL: <http://www.elftown.com/eye%20tutorial1>. 1, 4, 6
- [TBSR04] TSANG S., BALAKRISHNAN R., SINGH K., RANJAN A.: A suggestive interface for image guided 3d sketching. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (New York, NY, USA, 2004), CHI '04, ACM, pp. 591–598. URL: <http://doi.acm.org/10.1145/985692.985767>, doi:<http://doi.acm.org/10.1145/985692.985767>. 3
- [TH09] TAELE P., HAMMOND T.: Hashigo: a next-generation sketch interactive system for japanese kanji. In *In the Twenty-First Innovative Applications Artificial Intelligence Conference* (2009), IAAI '09, ACM. 3
- [UH07] URBINO L., HENRY D.: *Art Recreations: Being A Complete Guide To Pencil Drawing, Oil Painting, Watercolor Painting, Grecian Painting And Many Others.* Kessinger Publishing, LLC, 2007. 9
- [WP88] WOLF D., PERRY M.: From endpoints to repertoires: Some new conclusions about drawing development. *Journal of Aesthetic Education* 22, 1 (1988), pp. 17–34. URL: <http://www.jstor.org.lib-ezproxy.tamu.edu/2048/stable/view/3332961>. 1
- [WSP05] WANG B., SUN J., PLIMMER B.: Exploring sketch beautification techniques. In *Proceedings of the 6th ACM SIGCHI New Zealand chapter's international conference on Computer-human interaction: making CHI natural* (New York, NY, USA, 2005), CHINZ '05, ACM, pp. 15–16. URL: <http://doi.acm.org/10.1145/1073943.1073946>, doi:<http://doi.acm.org/10.1145/1073943.1073946>. 3