Alice Chau
Neil Chungfat
Shirley Hao

Calvin & Hobbes: An Interactive 3D Gallery

Proposal | Abstract | Goals | Design | Implementation | Individual Contributions | Lessons Learned | Acknowledgements | References

While the original purpose of many computer graphics has focused mainly on creating images which seem uncannily life-like, attention has recently been focused on creating non-realistic looking images. This technique, known as non-photorealism, has been the subject of much recent interest and development. As a result, cartoons and other similar artforms that were traditionally created by hand are now beginning to appear in the context of computer graphics. This project aimed to investigate these non-photorealistic techniques and to utilize those that were most fitting to model the Calvin and Hobbes comic strip. This would allow for faithful recreation of the original look and feel of the comic, while adding another dimension of visual realization to the traditional 2D comic strip. Our project also involved, in no small way, exploring and implementing the various types of character and scene modeling techniques available.

Introduction

The film strip "Calvin and Hobbes" was created by Bill Watterson in the 1980s and first appeared in the Universal Press Syndicate on November 18, 1985. Based on the relationship and adventures of Calvin, a six-year old boy, and Hobbes, Calvin's stuffed tiger, the cartoon amused readers of all ages until December 31, 1995, when the strip was retired. Watterson was against commercializing his popular strip, and consequently, no commercial animated version of Calvin and Hobbes was ever created.

Computer graphic artists once aimed at creating realistic models. Recently, however, efforts have aimed at creating non-photorealistic models, a genre which includes cartoons, comics, and other similar art-forms. Sugano Yoshinori, producer of the recently released animated film *Princess Mononoke*, believes that graphics, as an art form, aims to convey ideas through images. He claims that it is the interpretation of the ideas that is more important than how similar a given scene reflects its counterpart in the real world. He therefore states, "I think that computer graphics and non-photorealistic technologies offer us chances to create images which are actually more real to viewers than those created using photorealistic methods." (Yoshinori, 1999) Watterson was able to convey some of his beliefs to his readers by way of his comic strip. It is therefore our responsibility that the ideas conveyed by the original comics remain salient after our conversion of his work into the realm of 3D.

This project requires the use of several of the topics discussed in class from the perspective of an end user. For example, though we did not actually implement an algorithm for texture mapping, the use of this technique was of central importance for creating snowy fields in our scenes. Other issues that were important to deal with include lighting and camera angles.

Goals

We originally planned to model three stills (see Figure 1) and two full-length animations in 3-D for our project. Halfway through the project, we decided it would be more realistic to cut one of the longer animations down to a simpler, shorter animation sequence. Our central goal was to maintain the look and feel of the original 2D comic strip, while adding another dimension of complexity to the characters.

To combine the different stills and animations into one product, we wanted to design an interactive interface to allow users to browse our gallery. Since snow is the connecting element in the strips we chose, we decided to model a snowy world where the user can virtually walk around and choose to examine different objects related to each scene, such as a sled or a snowman. Upon selecting an object, the user will be able to view the 3D rendition of the comic or animation sequence in a separate window.

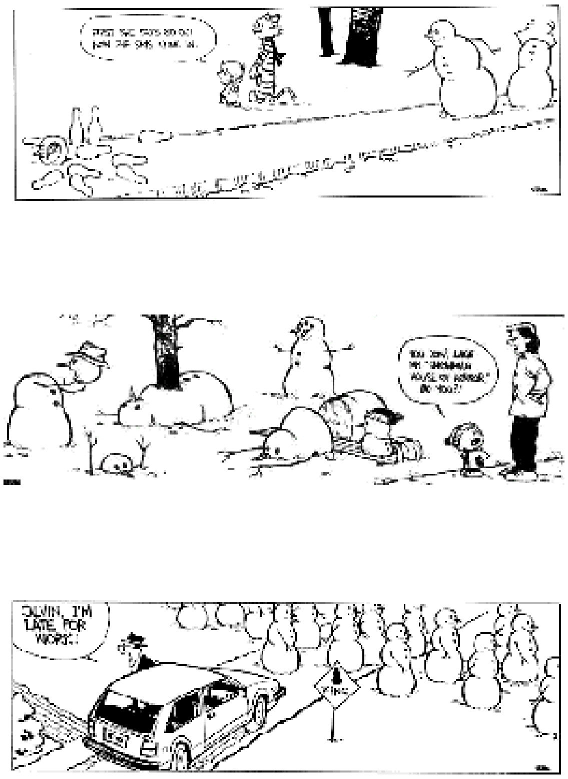

Figure 1: The three modeled stills.

Design

Using a well-known comic strip as the subject of our project presented both advantages and disadvantages. On the positive side, our scenes required very little planning in regards to storyboarding and the layout of objects. However, this also proved to be a constraint on our freedom to move around objects or make major modifications

Figure 2: The 2 strips used for animations.

around objects or make major modifications to any of the scenes. In addition, because the characters are so well-known, great pains were made to create accurate and recognizable models of Calvin and Hobbes.

The scenes modeled were carefully chosen, so as to limit the complexity of the project. For example, animations involved only the Calvin and Hobbes characters, thus allowing us to concentrate our efforts on modeling two quality characters. In addition, the sole use of winter scenes allowed a few simplifying features for the characters, such as the hat that covers up Calvin's somewhat unrealistic hair.

As the basis for our animations are a series of panels, key frames were essentially given to us. However, it was dependent upon us to decide how to segue between these frames and manipulate the camera views in a reasonable fashion. We decided early on not to provide voices for the characters, the reasoning being that everyone has a different mental idea of what Calvin's voice should sound like. Instead, we chose to place text bubbles that appear in key frames and then disappear as the camera shifts.

In order to provide the interactive world for the user to navigate through and pick the stills and animations, we had originally planned to use a programming package to make the world from scratch. Essentially by substituting the buttons on a toolbar with 3D objects, we wanted the user interface to be interesting and relevant to our comic.

Implementation

As none of the group members were very familiar with 3D Studio Max, our first task was to overcome the learning curve as quickly as possible so that efficient modeling could begin as soon as possible. Modeling issues arose early on and continued to emerge throughout the evolution of this project. These included many valiant attempts to model objects and characters using various modeling techniques, many of which were later discarded for one reason or another.

Character Modeling

A major goal of this project was to successfully model and render Calvin and Hobbes into three-dimensional space. Meeting this goal, while seeming straightforward at the onset of the project, did not fail to generate enough unforeseen problems to make for a lengthily frustrating, though ultimately rewarding, implementation process. Originally, the work effort was to be divided so that Neil would model Calvin, while Shirley would model Hobbes. We had planned to complete the character modeling by the end of the second week, in anticipation of the extended amount of time that the animations would require. However, very shortly into the modeling process, it quickly became apparent that these tasks would take us far longer than we had originally projected. This realization was due in no small part to the paltry modeling and non-existent 3D Studio Max experience that each of us had. In all three cases, we had only modeled before in Problem Set 3, and 3D Studio Max proved to be an altogether different matter from Inventor. Experimenting with 3D Studio Max allowed us to become familiar with its interface, but this short introduction did not necessarily give us the opportunity to find the most powerful options of 3DSM that we could exploit. More importantly, perhaps the first and most challenging difficulty that we encountered was how to approach the modeling. Our lack of experience put us at a loss as to how the characters should be built.

Our first attempt was to use primitives to create the basic shape of the character, and then to use mesh deformation to manipulate the plain geometric shapes into the familiar features of the kid and the tiger. Although this would have considerably decreased the time spent on modeling, we also found that we had much less control over manipulating the appearance of the characters that we expected; it did not give us the degree of command over the objects to bring the models to the accuracy that we wanted.

We next tried to use patches to make the heads, the most recognizable and therefore most important part of the characters. However, we found that it was quite difficult to manipulate the patches in 3-D space just by judging with the eye and comparing with the 2-D sketches, so we progressed to using splines as guidelines for patch placement and manipulation. By placing a side view of the character as the background in the side view window, and a frontal view of the character as the background in the front view window, we were able to lay out the splines so that the patches could be placed and manipulated more accurately. (see Figure 3)

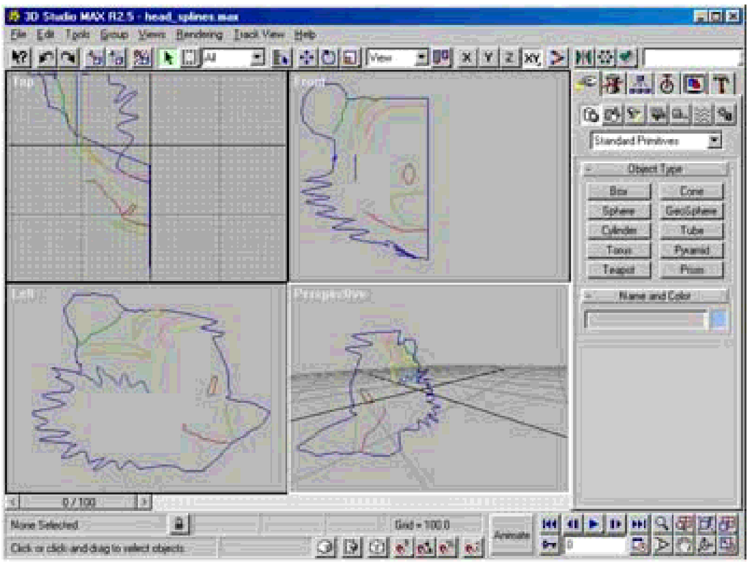

Figure 3: Layout of splines used in modeling Hobbes.

This was done by drawing a perfectly straight vertical spline in the front view, with height equal to that of the picture. The spline was "doubled over" so that it began and ended in the center of the picture. Then, in the side view, the horizontal direction was locked. Adding enough points to (refining) the spline necessary for an accurate capture of the features of the picture, we moved those points (and thus the spline) to the outline of the picture. The same process was done in the front view, but the spline was only moved to the left side of the picture because only half the character was being modeled. The other half would eventually be mirrored and welded to the original. The spline was now deformed so that it captured the profile of the character. Additional splines were added by the same process to outline the features and other necessary structural details within the face. Ideally the splines would have given us a 3-D skeleton to work with. However, the realization that Bill Watterson does not draw like a machine soon became a significant issue. Thus, even though the splines from the frontal and side views gave an accurate depiction of the character, it was nonetheless faithful only to that particular single sketch. The subtle details that make each of Bill Watterson's drawings unique were highlighted by the fact that in the 3-dimensional perspective view often came out unexpected because splines did not match up, or looked just plain strange. Because the one-dimensional splines were too difficult to manipulate in perspective view, we decided to resolve these problems in the patching stage; the splines would at least give us a general idea of where and how to lay the patches. Unfortunately, the patches soon proved difficult to use; once they became welded together, they gained the restriction that only one texture could be assigned to it. This was a problem because, without welding, it was almost impossible to align the patches perfectly. However, we needed to be able to assign different texture maps to different parts of the character. Patches were also more difficult to control than we had anticipated, and consequently left us disappointed at our intermediate results.

It was after these discoveries that we switched to using NURBS surfaces. (see Figure 4) We still kept the spline layout, but instead we generated an additional "helper" model made simply of a single primitive roughly in the shape of the head. The primitive was converted to a NURBS surface (CV surface), and, as in the case of the splines, the surface was refined so that additional vertices were added to the degree that we gained enough control over the surface as we desired. We very quickly learned the importance of refining rows/columns as opposed to refining them after encountering some very unfortunate and costly mishaps. NURBS were much easier to manipulate than patches and gave an amazing degree of flexibility and control. We had finally found a technique which would allow us to attain the reproduction accuracy that we desired.

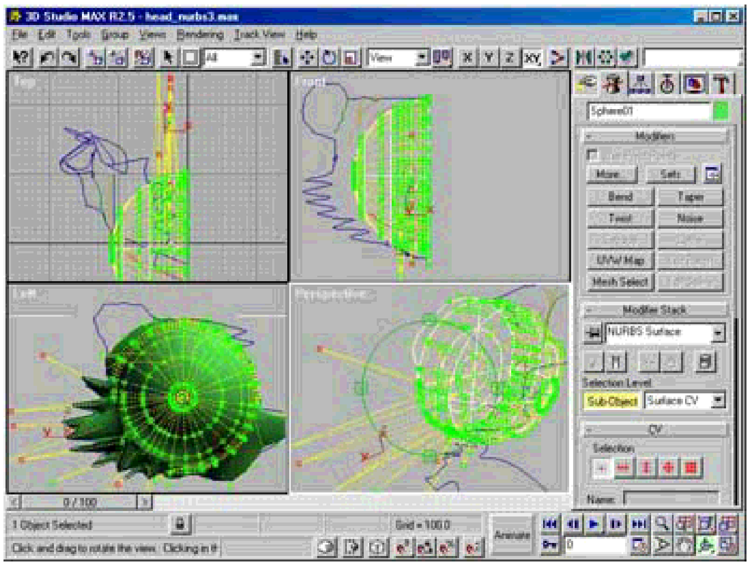

Figure 4: Intermediate stage in modeling Hobbes's head using NURBS surfaces.

The bodies of the characters were done in much the same way as the heads, using spline layouts as skeletal guides. Primitive cylinders were then placed appropriately and converted to NURBS surfaces, since the arms, legs, and trunk of the bodies were basically cylinders. The presence of the control vertices allowed us to manipulate the bodies where needed.

Since the actual texture required of the characters was minimal (Calvin and Hobbes are drawn with little definition of individual strands of hair and fur), the only major texture mapping required was the stripes on Hobbes's body. This was done by creating the bitmap using a 3DSM plug-in called 4D Paint. We exported the (almost) finished model to 4D Paint, which then allowed us to paint the features that we wanted directly onto the model itself. This approach gave us the flexibility of re-painting the model for the different scenes, when, for example, Hobbes's stripes are placed in different locations.

After the models were finished, we created and scaled appropriately a Biped for each model. The models were then fit to the Bipeds using Physique. (see Figure 5) Both of these tasks were accomplished using the plug-in Character Studio 2.1, with the goal that the models would be much easier to animate. As with so much of the character modeling process, this part was more time consuming than expected. Difficulties arose in aligning the model to the Biped accurately so that when the skeleton moved, the model ("skin") was deformed in the expected fashion. Hobbes's long trunk and tail proved especially awkward because of the implied flexibility those two parts of his body are given in the 2-dimensional world of comic strips. Since the Biped was limited in the number of spinal and tail joints that could be assigned, we needed to place and bind the vertices of the skin to the joint locations that were critical to our animations.

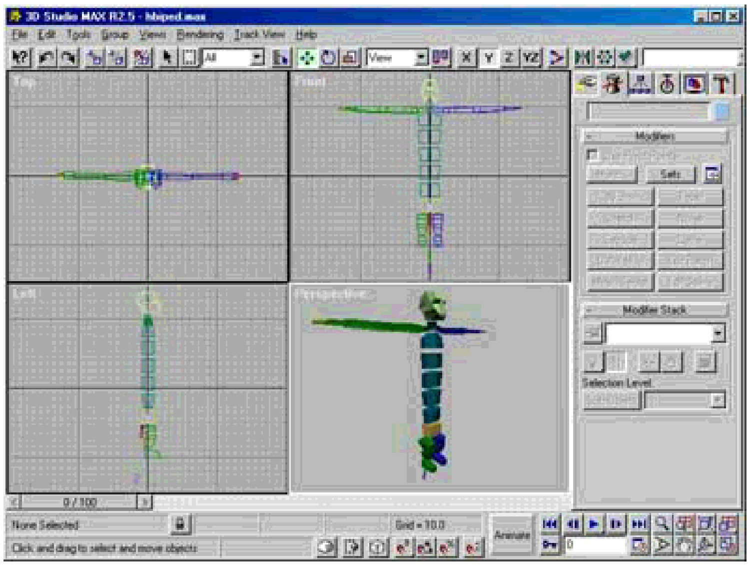

Figure 5: Biped model for Hobbes

The process of creating the Calvin and Hobbes models was, without a doubt, long and frustrating. Much of this was compounded by unfamiliarity with 3D Studio Max, a tendency for the program to crash unexpectedly (probably a result of the application's demanding system requirements), and files which were inexplicably corrupted or deleted as a result of the crashes. In the end, though, it became a very interesting learning exprience, since we eventually went through a great number of the major modeling techniques applied to character modeling. Although reading tutorials and texts provided general ideas on how to approach the modeling, it would ultimately take time, and lots of it at that, in order to really understand and progress through, the modeling process.

Scene Modeling

The snowmen in the scenes were created using primitives, in some cases, and lathing splines in others. Bowling pins, hats, and sleds were created in a similar fashion. Snow was produced by creating a bright white texture with a small amount of fractal noise. (see Figure 6) Backgrounds were also mapped using images of sky and clouds. Positioning the objects was not difficult in itself, except for matching the perspective of the imitated strip.

Figure 6: 3D stills.

Animation
The animations were purposely chosen to be relatively simple. Character studio was a great help in animating the walking motions of Calvin and Hobbes. The motion of the sled going down the hill and narrowly escaping trees was done by having the sled follow a path traced out with a spline. As the spline contains some inherent directional information, it required a few tries to get the sled to go down the hill facing the correct direction. Clever use of camera angles were used to get around potential pitfalls, such as hobbes setting the sled down on the snow. Combining different camera views and animating the characters' expressions were other difficulties encountered.

Interface
At the onset of the project, we had hoped to use either Java 3D, a 3D graphics application development kit, or FLTK, an interface builder with graphics capabilities, to program the main navigational interface for this project. These toolkits, however, were unfamiliar to us and required considerable time to master. They also seemed to have limited features regarding navigation through the interface. Upon discussion with our TA, we discovered that some software made by Cosmo Software, namely Cosmo Worlds, the VRML authoring and publishing program and the accompanying Cosmo Viewer were more appropriate for our tasks.

VRML, similar to HTML, is an open language developed for the web. Cosmo Viewer allows users to navigate through a VRML 3D world. (see Figure 7) While the user walks around the world, various actions are triggered through proximity to certain objects (sounds, rudimentary animations) or through explicit mouse clicks (animation movies, jumps to urls). The 3D objects, navigational style, triggers, and animations can be made using Cosmo Worlds. We found this application to be more suitable to our project goals since it already has most of the features we wanted built in.

In the actual implementation of the main navigational scene, several interfacing factors such as restriction to the scene and triggering became important. Since our world is limited, we had to restrict the user to only look around and trigger actions in a specific part. By only allowing users to walk instead of fly, and building mountains as borders, we were able to appropriately limit the user's navigation.

The triggering mechanism of the world is perhaps the most important element in our interactive scene. We had to make sure that the user was constantly engaged and interested in our Calvin and Hobbes world by encouraging to explore all the objects in the scene and providing audio and visual feedback when they have reached an important part of the scene. To accomplish this, proximity sensors were set for all five of the main subscenes (the 3 stills and 2 animations) so that music would begin to play when the user was approching the scene. Rudimentary VRML animations of a certain part of the scene or object would also begin to provide a visual prompt for the user to explore this scene further. If the user then decides to click on some part of the scene, the touch sensor is triggered and another window with the appropriate animation or scene will be displayed.

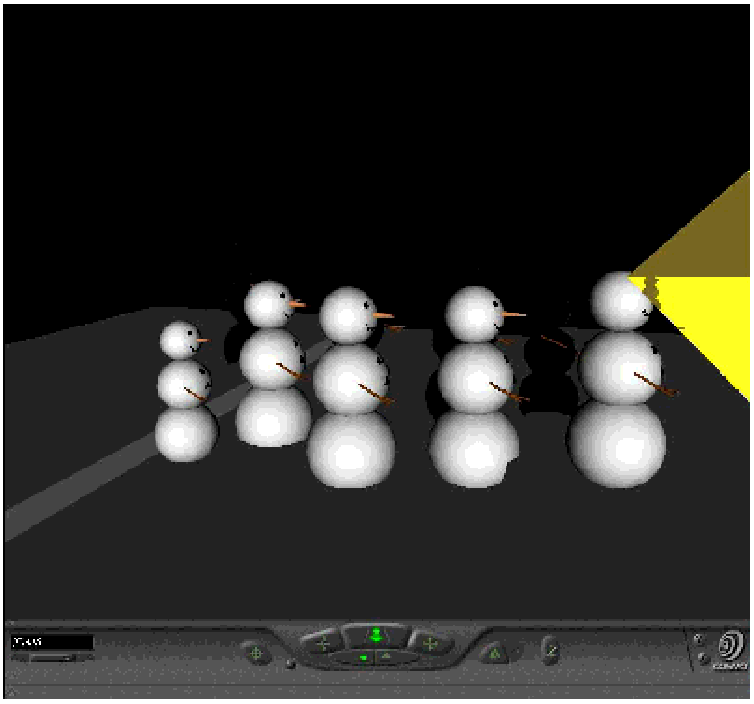

Figure 7: An example of the Cosmo Viewer navigational interface for an incomplete scene

One main technical concern of this navigational user interface is the rendering speed of the scenes. Since most of our scenes are pretty complex, they take considerable time to generate and would interfere with the interactivity of our application as a whole. Our world most render in real time so the user is not stuck waiting for graphics to appear. To accomplish this, scenes and objects that were a considerable distance away from the user's current location are rendered under their low level of detail mode. This entails representing snowmen as a series of blocks rather than spheres to lower the polygon count. As the user approaches the scene, the details return to the scene and exploration continues as normal.

Another technical challenge we faced was exporting 3D Studio Max objects into Cosmo Worlds. Since all the scenes were generated in 3D Studio Max, they all had to be exported as VRML. Because of the complexity of the scenes the programs apparently could not handle exporting max files to VRML, resulting in system crashes. To accomodate, we had to basically split up the scenes and export groups of objects instead of entire scenes. Then, we had to compose the scenes again in Cosmo Worlds and add back certain details such as the snow covered ground. In the end, scenes were stored as separate world files and the main navigational scene referenced these inline files, providing a nice organization for the main scene and also reducing its size.

Individual Contributions

Shirley and Neil were responsible for the modeling the stills and animations in the project. Shirley's efforts went into creating both Calvin and Hobbes, as well as the bowling scene, while Neil contributed the remaining two stills as well as the environment for the sled animation. Alice researched various graphical user interface options, eventually deciding on using Cosmo Worlds for development of the VRML world. She then created the navigation scene for use in this project by integrating the stills and 3D objects from Shirley and Neil's work and adding various triggers throughout the world for a more interesting user experience.

Lessons Learned

Shirley and Neil quickly became very familiar with a limited subset of the plethora of tools available in 3D Studio Max. By experimenting with patches, NURBS surfaces, meshes, and texture mapping in creating the models, the advantages and drawbacks of each technique were elucidated. The project also required experimenting with different kinds of light sources, cameras, and animation techniques. Due to some instability inherent in running 3D Studio Max from Windows 95, the importance of saving often and backing up our files was a lesson learned the hard way. Overall, Shirley became the expert modeler of the group, while Neil learned the various facets of the animation process.

On the other hand, Alice learned about some of the 3D graphical user interface tools available online and the advantages and disadvantages with each of these packages. She became very familiar with the chosen tool, Cosmo Worlds, and learned a great deal about the capabilities of VRML. By experimenting with different navigational styles, viewpoints and triggers, she also learned how to write a flexible but robust user interface to allow the user freedom to navigate in the world without reaching the bounds of the world and keeping the user interested in the world. She also discovered a great deal about optimizing the world by using low level of detail and scene organization.

Acknowledgements

We must, of course, acknowledge Bill Watterson, the brilliant mind behind Calvin and Hobbes. For years we have enjoyed the wit and cleverness of his comic strip, and it was exciting for us to be able to get to know his work more intimately through this project.

In order to become familiar with 3D Studio Max, we used online tutorials offered by 3D Cafe, which were found at http://www.3dcafe.com/asp/tutorial.asp. The tutorials were very useful in giving a basic introduction to the 3D Studio Max environment, as well as a general idea of the various types of tools that 3D Studio Max offered.

As some objects proved to be quite difficult to model relative to their overall importance of the scene, some were borrowed from sites that offered free models. The car in the "Snowman Crossing" scene was taken from http://www.maxnet.ru/aks/models_R.html The color was changed from black to red, and the chrome texture that was applied to the car was removed to make it appear more like comic-like. Some of the trees in the scenes were also taken from http://www.3dcafe.com/asp/plants.asp. Cloud bitmaps used for background textures were obtained from http://www.commerce.digital.com/palo-alto/CloudGallery/see_clouds.html.

Acknowledgements must also be paid to Cosmo Software for providing the Cosmo Worlds and Cosmo Viewer software integral to our project.

We must also not forget to mention the helpful support and advice that we received from our

Damian Isla, the teaching assistant assigned to our group, and Professor Seth Teller for his helpful responses to our proposal and ideas.

References

Elliott, Steven D, et al. Inside 3D Studio Max: Advanced Modeling and materials. New York: New Riders Publishing, 1997.

Peterson, Michael Todd. 3D Studio Max Fundamentals. New York: New Riders Publishing, 1998.

Watterson, Bill. The Calvin and Hobbes The Lazy Sunday Book: A Collection of Sunday Calvin and Hobbes Cartoons. New York: Andrews McMeel Publishing, 1989.

Watterson, Bill. The Authoritative Calvin and Hobbes. New York: Andrews McMeel Publishing, 1990.

Watterson, Bill. The Indispensible Calvin and Hobbes. New York: Andrews McMeel Publishing, 1992.

Yoshinori, Sugano. Manga and Non-Photorealistic Rendering. ACM SigGraph Newsletter. February 1999, Volume 33, No. 1.
http://www.siggraph.org/publications/newsletter/v33n1/contributions/Yoshinori.html

---