

ROCHESTER INSTITUTE OF TECHNOLOGY

A Thesis Submitted to the Faculty of
The College of Imaging Arts and Sciences
in Candidacy for the Degree of

MASTER OF FINE ARTS

DIGITAL CAMERA, AN INTERACTIVE TUTORIAL

By

Roger Martínez i Colom

NOVEMBER 1993

APPROVALS

Adviser: Robert P. Keough

Date: 1-31-94

Associate Adviser: Jim Ver Hague

Date: 1/26/94

Associate Adviser: Mark Collien

Date: 1/18/94

Chair, Graphic Design: Robert P. Keough

Date: 1-31-94

I, Roger Martínez Colom, hereby grant permission to the Wallace Memorial Library of RIT to reproduce my thesis in whole or in part. Any reproduction will not be for commercial use or profit.

Date: _____

TABLE OF CONTENTS

❖ Preface

- About the Author (2)
- The Proposal (3)
- Historical References (4)

❖ The Project

- The Design Process (8)
 - Previous Considerations (8)
 - Technical Requirements (8)
 - User Interface (10)
 - Structuring the Application (11)
 - Selecting the Contents (12)
 - Deciding the User Interface (13)
 - Deciding the Aesthetic Appearance (15)
 - Building the Application (15)
 - Producing the Imagery (15)
- Evaluation Process (21)
- Concluding the Application (22)

❖ Conclusion

- Evaluating the Tutorial (25)
- Future Developments (25)

❖ Appendixes

Scripting Reference (28)

Flowchart (30)

Visual Portfolio (31)

Bibliography (44)

Preface

ABOUT THE AUTHOR

This section provides the reader with a brief profile about the author of this thesis. This profile consists of a short personal portrait, and a wider description of the education and background that preceded the current Master's degree. Even though a profile about the author is not needed, it may help in understanding how and why things have been done the way they are now.

Roger Martínez Colom was born in June 1967 in Girona, a small town close to Barcelona, Spain. Even though he has always been attracted by science, he studied his Bachelor's degree in Fine Arts, in the School of Fine Arts in Barcelona. This Bachelor's degree has a length of five years, with the last two specialized in graphic design.

The School of Fine Arts started experimenting with the use of computers applied to design just a few years ago, and he was given a chance of joining a team that tested this new field. This provided his education with a wide background of computer use. Some of the systems used in the program were Apple Macintosh, DOS, Amiga, and VMS related systems. Working experience was mostly based on teaching the basics of computer graphics, especially interactive computer graphics.

At the end of the Bachelor's degree, he applied for some scholarships that could allow him to study in USA. The Catalonian government in Spain awarded him a scholarship, and he started the Master's degree in Computer Graphics Design at the Rochester Institute of Technology. He specialized in interactive media design, increasing his

knowledge on the field.

This thesis is the culmination of his educational experience, and it supposes one of the best pieces of his portfolio.

THE PROPOSAL

The purpose of this thesis is to explore the possibilities of computer graphics applied to the creation of instructional projects. Chosen the topic of "basic photography", I will produce an interactive application that will teach several techniques used when learning photography. The project, fully interactive, will combine several media (computer generated images, acquired images, sound, etc.), all controlled by the computer. The main goal is to develop a proper user interface which allows any person to utilize the application.

The original proposal, cited above as it had been turned in to the Department, was slightly transformed. Rather than changed, it was extended. The first idea was then applied to a specific topic: the development of an interactive tutorial that shows how to use a camera, in this case the Nikon FM2. In the tutorial, basic knowledge of photography would be combined with the tips on how to operate the camera.

The application should show several processes that help the user understand easily how to get more advantage of the camera. Printed manuals or operation books provide users with basic knowledge and technical information. New

interactive applications, though, can enhance this knowledge with demos in a more friendly and personal way.

HISTORICAL REFERENCES

When we compare the computer age to any other period in the history of humanity, it is amazing to see how fast it has evolved. With approximately half of a century of life, computers have grown faster than anything before. They evolve so fast that the usage of a specific computer is expected for just a few years until is replaced for a much better model. Computer graphics are much younger than any other field of computer science. Technology provided us with graphics applications around thirty years ago.

In the last decade, a specific field of computer graphics acquired more and more popularity: multimedia graphics. Called also interactive media, interactive graphics, etc., this field joins image, sound, text, and others, using programming. The media is becoming every day more and more attractive. Recently, during the last months, the technology has made available a set of very powerful computers for a very affordable price. An increasing number of users promises this field to be the revolution of the nineties.

When technology is available for a majority, designers should start asking themselves what they could do in order to get the most advantage of this technology. Many applications have already been found: commercial communication, educational, etc. Among them, education seems to be using this technology on a big scale. The capability of learning at

your own speed, upon your personal interests, has been making educational multimedia more attractive.

Until now, educational multimedia has been just experimental. Equipment was too expensive and slow. It was not profitable to produce interactive applications. Schools could not afford to purchase the proper equipment. Therefore, because of no demand, the making of these applications was limited. Computers are becoming more and more available today, and this field will very soon be of extreme importance for education.

Despite of its recent creation, many multimedia projects have been developed. At the very beginning, just a few years ago, experiments were produced with no methodology. The field was so young that designers, more than reasoning, had fun creating attractive graphics. The learning was completely autodidactic, which means slower and unsure. Schools are just introducing new programs of study, and soon designers start having the ability of justifying their knowledge. Only now, when technology and education have mixed together, are we ready to produce our best in multimedia design.

The Project

A methodology is an organizational way of analyzing and structuring information. It is an attitude used to pursue a goal. Each one of us has unconsciously--and consciously--a way of executing different actions, which range from very common behavior to complex creation.

The methodology used in the developing of the application was set to be strict, and it had to follow a sequence. Computer graphics are a blend of art and technology. Usually, the artistic methodology is very flexible and personal. The effects of this "free" artistic methodology are unwanted when applied to multimedia. The results bring applications to unexpected consequences. Multimedia educational projects are very valuable when we control them precisely, when we can predict all the results. Therefore, an accurately planned methodology would produce the correct result.

This section will describe all the steps involved in the creation of this application. It is divided into two main parts, which are sequential. The first one is the design process itself, and it explains all the steps involved in creating the first version of the application. The last one is the modification or correction of the application upon evaluation.

THE DESIGN PROCESS

To develop a first running version of the application, I divided the process into three sequential steps. The first one analyzed the objective pursued, and listed all the considerations involved in its creation. Then, the project was given a shape: information was structured, and the sections of the application were stated. Finally, using MacroMind Director, a computer model was built and run for the first time.

PREVIOUS CONSIDERATIONS

The previous discussion of the technical aspects, the possible users and environment that condition the project is fundamental. Creation is an action that involves many conditions. The act of creation without conditions, completely free, is extremely hard. The more conditions we have, the faster we envision a solution to a concise problem.

TECHNICAL REQUIREMENTS

The first step was to limit the technical capabilities of the application. The main objective of the thesis was to produce an application that was useful as an educational

tool.

The computer systems most extended today as educational tools are PC's and Apple Macintosh. Both systems are equitable, they both offer similar possibilities. But here, the technical prerequisite was to use the systems available at RIT, more precisely in the Graduate Computer Graphics Design lab. Thus, the envisioned system would be Apple Macintosh. Generalizing, the computer that the average Macintosh user may have was supposed to be any MacII with 8 bits of color depth, and a screen size of 640 by 480 pixels. This condition defined the screen size of the future application to a maximum of 640 by 480. Regarding the file size, I anticipated the application file to a maximum of one high density disk, if possible. In general, a "Superdrive" (1.4 Megabytes floppy disk drive) is expected in every Macintosh. I decided that if this had to be an educational tool, it would possibly have to be distributed to the public (note: when planning this thesis I had no intention at all of distributing the product). The most comfortable and affordable way of handing out software is still a floppy disk.

Concerning the software involved in the creation of the application, a "hypermedia" program should be used, since the idea involved static graphics, animations and interactivity. Three the possibilities that should be explored were Hypercard, Supercard and MacroMind Director. From the beginning, though, Hypercard was discarded due to its poor color managing capabilities. Considering my personal skills with either Macromind or Supercard, I selected first Supercard. I handled Supertalk (Supercard's programming language) more than Lingo (Director's programming language). Other software would be used as a complement to create more graphic materials for the thesis, like Adobe Photoshop, Stratavision 3D, etc.

After analyzing all of this, the project was technically set up: an Apple Macintosh interactive application, done in either Supercard or Director (later on, I chose Director, due to its power to manage animation), with a size of maximum 640 by 480 pixels and 8 bits of color depth, and a file size of about one high density floppy disk (if possible).

USER INTERFACE

Now, the problem was to humanize the project: make it appropriate and easy for the target user. Therefore, I needed to define who was the target user. The idea itself was to develop an instructional tool for anyone who was planning to purchase the Nikon FM2 camera. This instructional tool would teach the basic steps on the camera's operation, as well as fundamental photographic concepts.

The profile of the ideal user was set not to be very experienced on the use of computers. Even though, the basics on how to handle a mouse or touch-screen would be assumed, so would be the intuitive process of moving through different screens. More than that, the application itself would have to be clear enough to allow free navigation with no need for help. Furthermore, the use of technical words or uncommon objects on the screen was to be avoided. Usually, inexperienced users feel hopeless in front of computer vocabulary. The navigational system would have to be as natural as possible (something like "what you see is what you can click").

In order to provide the overall project with consistency, I decided to use color-coded objects to point to clickable objects. In the final application, all the objects (text, buttons, arrows, etc.) in red are clickable. These codes are

described in the next section "Structuring the application."

Another user-interface decision was not to overload the screen with information. Excessive amount of buttons, text, graphics makes the user feel impatient and uncomfortable. Aesthetically pleasant graphics provoke better responses in the user.

Probably, the main part was to decide the general metaphor of the project. Here, I decided to make the navigational system look realistic. Usually, we are so used to working with computer programs, that their abstract organization is more than familiar to us. Pop-up menus, dialog boxes, etc., are specifically designed for the computer media, and has no relation to any object we find in the real life. If we had had no contact with computers before, we would not really know how to act in front of a pop-up menu, or similar objects.

For that reason, I decided to create an interface closest to reality as possible. This may be called today "virtual reality." Virtual reality emulates real life. It is the most natural interface, and it demands almost no knowledge from the user. As an example, if I had to control the camera, instead of using buttons labeled as "set shutter speed" or so, I would let the user handle the real image of the shutter dial. Whatever you would do on the screen is exactly what you could perform on your Nikon FM2.

STRUCTURING THE APPLICATION

A general idea of what the thesis would be like was now in mind. The process now was to select the contents of

the project. The amount of information delivered to the user was key to this project, since it is instructional. Later on, when this was done, I concentrated on how to use the software to properly deliver the information (user interface). Finally, when everything above was clear, I chose the appropriate aesthetic appearance of the application. All the three steps are described below, in this section.

SELECTING THE CONTENTS

The amount of information delivered is decisive. In an instructional project not only we have to know what to say, but we also must control that this is not too much (or not enough) information. The nature of the thesis was to be a tutorial. Then, I thought that an excessive amount of details would make the project confusing. The application itself should teach the basics on how to operate the camera, and how to get the best advantage of it to create good photographs. Reviewing the actual operation manual of the camera and some books on basic photography, the contents were divided into three main sections:

Section 1: Called "nomenclature," it would visually point to every part of the camera to name it and give its function. This would let the user recognize visually every switch, button or dial in the camera.

Section 2: Called "basic operation," this would teach the very basics on how to operate the camera. The topics contained in this section are: installing the battery, mounting the lens, installing the film, setting ASA/ISO, basic exposure, and rewinding the film. All these topics are found in almost every operation book that accompany cameras (including the first section).

Section 3: "Controlling exposure" was the last part. This was the most attractive part. In a very visual way, the user would learn by practicing how to get the best advantage of his camera. The main topics covered here are: focusing, setting shutter speed, setting aperture, determining exposure, displaying exposure and exposure in special cases.

After all the contents were structured, research was made to fulfill the objectives of this section. The last section was completed by researching books on basic black and white photography, and experimenting with real film.

DECIDING THE USER INTERFACE

The project was meant to use graphics (all the graphic elements displayed on the screen, including text), clickable buttons, sounds and animations for delivering the contents to the user. The main issue here was how to properly organize and display all these elements so the user could learn all that was planned. All these elements had to be clearly separated. Anyone should be able to recognize a button as a button. In every section, the user should know what to do to continue viewing the tutorial.

A visual code needed to be created. Part of this code was: first, how to recognize each element (graphics, buttons and so); second, how to navigate through the project; and third, how to make it visually stimulating so the user would move around by intuition.

Elements were to be understood by the position they were placed, and their color, shape and size.

Spatial Organization

The screen was divided into two main parts. Almost all the principal buttons would be placed in a rectangular area on the right side of the screen. This area would only contain the subtopic buttons, and the main topic buttons plus the exit button. The main topic would be displayed as a title on the top of the left part of the screen, as a heading for the information. This would be only placed on the left area. Even though these areas were not mixed, sometimes additional control buttons were positioned on the left side.

Organization of the Visual Appearance

The colors of the buttons would be variable (brown for subtopics, blue for general topics, and green for the "exit" button). When a subtopic button was activated, its color would switch to red. The red color was then associated with buttons. Otherwise, if a button was not placed on the right side box, it would have to be clear enough to allow its recognition. Red-stroked arrows were used as alternative buttons.

Red color, after all, was associated with current elements, either buttons or graphics that needed to be understood anytime. In every screen, if something was red, it had to be read or could be clicked. Buttons had the feedback of a "click" sound. Once buttons were easily recognized, navigation would become intuitional.

Titles, subtitles and main text would be differentiated by using different text font, size and color. I also tried to keep the amount of information displayed on every screen balanced. I chose to never overload the screen with too much text, or graphics.

DECIDING THE AESTHETIC APPEARANCE

I first wanted the project to look visually different. I tried to avoid flat colored surfaces, dialog boxes and elements typically associated with computers. I chose to accomplish this by using a custom-made texture, and consistent colors. As for the texture, I planned to provide the right side panel with a metallic look, similar to the metallic parts of the camera. I wanted also to provide some elements with a three-dimensional look (by using shadows and embossing). The left panel, containing the main information, would have a soft texture that would not confuse its reading. The colors would be soft, ranging from a custom-made palette of browns and greens.

BUILDING THE APPLICATION

PRODUCING THE IMAGERY

It was time to actually start making the Tutorial. I chose to divide the whole process into two phases: first, to prepare all the graphics; then, to import them in Director, and to program the application by using Lingo.

Two kinds of images had to be used: digitized images of the actual camera, and computer generated imagery. Most of the computer generated imagery was simple, but some complex effects were created originally, and are listed below.

Creating a Custom Clut

The very first step was to create a clut or palette with the colors I was going to use. For space and speed matters, an 8 bit palette had to be built. The standard palette for 8 bit mode is usually what is called the "system" palette. It contains

a general representation of the major colors needed to reproduce a color image, and it includes 10 levels of gray. Since I was planning to use mostly black and white images to explain different processes, I found this range of grays very limited. Regarding color, the system palette was quite complete. But the gray levels were unacceptable. I needed at least 16 grays to make a gray scale picture look real.

Therefore, I started the process of elaborating a custom palette. The process was complex: first I edited and transformed the system palette in Supercard, because this program has more tools and lets you pick and blend colors easily. Once the palette was changed to what I wanted, I needed to save it in a format that could be understood by Photoshop. I figured out that the only way, was to capture the screen and save it as a Pict file (by pressing the standard Command+Shift+3 keys). From Photoshop, I opened the pict file containing the image of the whole screen, and I cropped it to the size of the clut.

Then, I had to save that Pict file into a clut file. Currently, there is no other way of doing this rather than



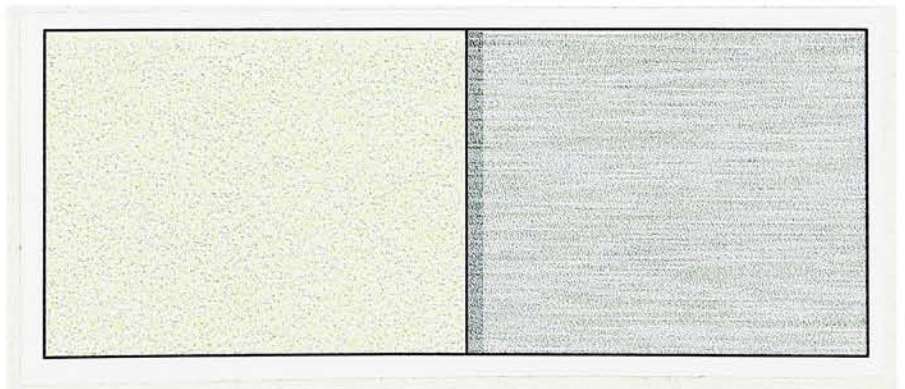
On the left hand side, the default system clut. Next to it, the custom-made clut. The main differences are a wider range of grays (at the bottom) and cream colors (at the top). The remaining colors are extracted from the system clut.

setting the RGB percentages for each color of the clut. What I did, is get the numbers of every portion of color of the captured clut by using the "eyedropper" tool in Photoshop, and write them down. Then, selecting the "color table" menu option, I built a new clut color by color (this is a very slow process), and save it as a clut. I had finally the 8 bit custom-made clut, which I used when saving all the graphics I had.

Creating the Background

Each of the two sides on the background has a different texture. The grainy texture on the left side is created just by using the Photoshop filter "noise" on a cream-colored background, with a setting of 32 (which controls the amount of grain).

The "metallic" texture on the right side is slightly more complex. I started by adding noise, just like before. Then, a "motion blur" filter with a setting of 30 expands every pixel horizontally 15 units in every direction, providing the background with the metallic appearance. Then, the color is easily changed to a gray metallic, instead of cream. The

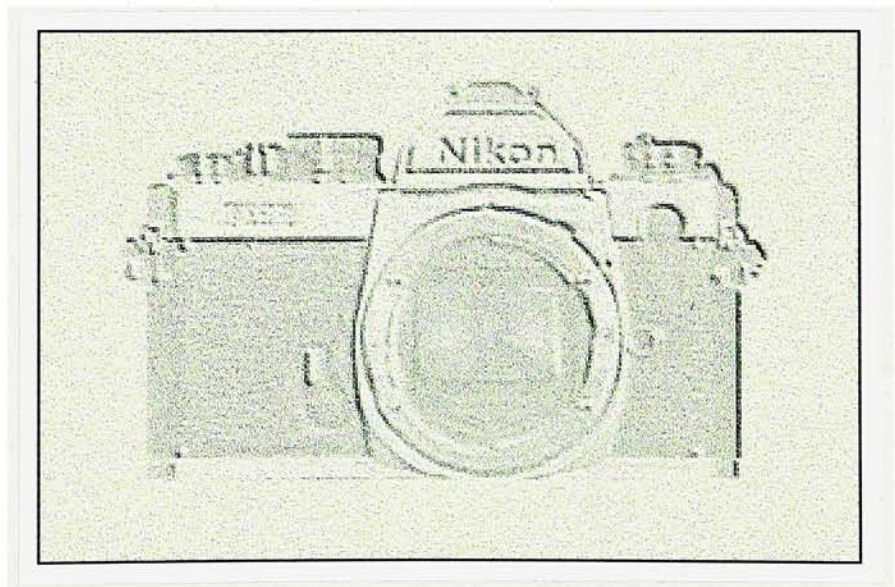


Here there is a sample of both textures for the background. The left side is just a grainy texture. The right one simulates a metallic surface. The shadow in the center helps recreating a tri-dimensional space.

three-dimensional edge that separates both sides is achieved by selecting the edge area with the select tool, and changing the brightness to darker.

Embossing the Background Camera

The embossed camera graphic was meant to be a separated cast member in Director, since it appears unevenly through the Tutorial. The problem in this step, is that placing first the scanned camera onto the background, and then embossing, would only produce a noisy image (since the grainy texture is also embossed). A clear embossed image is produced if we first emboss the camera on a clear cream-colored background, and then we apply the "noise." It is also important to previously reduce the contrast and increase the brightness, so the image of the camera is almost invisible.



Described above, the steps on how to emboss a graphic on a grainy background.

Effectively Integrating Graphics on to the Background

Every element is independent from the background. This means (in Director) that every title, button, text, does not contain the image of the background. Though, when placed on top of the background, it looks like it is part of it, because it is smoothly integrated (antialiased).

Even though there is an option in Director that lets you antialias automatically, the quality is not good enough, and it reduces the speed. There is a better way to keep antialiased images of the objects we plan to place on a background, and it is achieved in Photoshop by following this process:

First, set the current background to the average color of the desired background (let's say a cream color). Then, type the text in the desired color, by using the text tool and selecting the antialias option. Now, to make the background of the text disappear (so we can see it transparent in Director), we use the wand tool. We call the dialog box related to the wand tool, we deselect its antialias option and we set its tolerance to 1. Then, we select the cream background by clicking on it, and we choose "similar" from



On the left side, correctly antialiased graphic using the above described method. On the right side, default and non antialiased graphics (magnified to 200%).

the "select" menu. This will select all the cream color we wish to get rid of. Then, we just delete it. What we get, is an antialiased text that looks almost unreadable (because we have just deleted the current background). When, in Director, though, we place it on the grainy background, it looks perfectly integrated, and it saves memory and speed.

All the graphic elements in this project have been antialiased this way. In Director, the elements in the score have been selected all "background transparent."

Creating Effective Shadows

The process is similar to the one described above, and it is also accomplished in Photoshop. I started also by having a cream background (in this example, the color is cream). Then, a black image of any graphic we need to shadow is placed onto the background. After selecting that black graphic, we use the "gaussian blur" filter to diffuse it. With the "wand tool" set also to tolerance 1 and no antialias, we select the background again, and we choose "similar" from the "select" menu. This will select all the cream color that it is supposed to be the background. Once we delete it, we convert the graphic to 8 bit using our custom clut.

When, in Director, we place the real object on top of the shadow, it will look very realistic.



EVALUATION PROCESS

Once the Tutorial was finished, I ran it myself trying to find errors. I realized then I had made two different kinds of mistakes: execution errors and concept errors. After correcting what I found myself, I let several users test it (a few Mac experts and non experts also). Listed below are examples of the errors found when testing (in Director, since this was already the testing stage).

1-The animation, which was supposed to be stopped any time by pressing the button, did not work interactively. After trying different scripts, I found the best way to show it was sequentially, without letting the user stop it. The problem was found in the transition channel: dissolve effects conflict with interrupting scripts. I finally had to choose between interactivity, or visual appearance (I discarded interactivity in the animation). This was a score error.

2-The “click” sound did not work properly in any button. I found that the command “puppetsound” has to be reset (puppetsound 0) after it is played, at the end of the script. This was a scripting error.

3-The “Controlling Exposure” menu did not work properly. The computer did not match the wanted option. I discovered, after reviewing all the scripts, that the coordinates of the cast member were set to the center of it, instead of the upper left corner (this caused a misplacement of the menu). This was a cast error.

4-The “Exit” button quit when it was clicked. Instead, I figured it would be much better to restart to the animation. I thought, since this was supposed to be a continued tutorial, that users should just restart when finished, instead of quitting. This was a concept error.

5-The visual effects were inconsistent. I noticed that transitions used in opening sections were not all the same. They were all unified. This was a design error.

Other design errors, like the ones above, were corrected to provide the Tutorial with consistency. Once this process was finished, some other users tested the application again, and small errors still were found. As a simple comment, I realize that the best way of testing the application is during the thesis show. The show brings very different kinds of people that bring up different opinions and recommendations.

CONCLUDING THE APPLICATION

After the Tutorial was tested and corrected, I considered it finished. The last step, then, was to build a standalone, making it independent and original. The process, quite simple, is described below.

To build standalone in Director, we need to use the Player utility that comes with the program. This utility, used also as a driver for playing movies from other applications, allows us to run our movie independently from Director (which means that you don’t have to use Director anymore to run it). Choosing the option “Make a projector” I selected the movies I wanted to join as a standalone (in this case, just

one), and the project was compiled. In the finder, it appeared then with a projector icon.

The Tutorial was running alone, and the only step left was to change its icon. In Photoshop I had created my own icon, and I just applied it to the Tutorial file by selecting “get info,” from the finder, and pasting the new icon on top of the old one.

In addition, I used ResEdit to change the creator name and the version name, making it completely customized.

Conclusion

EVALUATING THE TUTORIAL

After I finished the Tutorial, I found it was a little different from the original idea, which was more complex. I realized that technical circumstances forced me to make it simpler. I also underestimated the real amount of time needed to complete it.

Acting as a self-critic, I see that the third part of the thesis (“controlling exposure”) is the closest to my first proposal. The idea of virtually changing the settings of the camera and immediately observing the results, was the main point of the thesis. Unfortunately, I see it was only maintained in the third part.

Though, I am satisfied that I accomplished the third part (which originated more comments and surprise from all the users), I consider the user interface very easy to learn (no help is needed). The aesthetic appearance is consistent and enjoyable. The fact that QuickTime movies occupy an enormous disk space made me change the technical expectations I first had.

If I had another chance to start over, I would choose from the beginning Director (instead of going through the process of discovering Supercard was not appropriate), and I would make the first two sections more interactive.

FUTURE DEVELOPMENTS

Multimedia offers enormous possibilities to education. I found that attractive (visually, interactively) applications

force more interest in learning. Virtual reality is still the center of attention. A future development of this project could explore more the possibilities of menu selection (make them more unattached from computer standards). Any attempt to create original projects applied to education should be considered for future development.

Appendixes

SCRIPTING REFERENCE

The scripting I used through the Tutorial is quite simple. Below, two different scripts that may be of interest to note:

```
--GO TO NEXT FRAME
puppetsound "click"           --PLAY THE SOUND "CLICK"
cursor 200                    --SET THE CURSOR TO BLANK
go frame (the frame + 1)     --GO TO THE NEXT FRAME
cursor 0                      --RESET THE CURSOR
puppetsound 0                --RESET THE SOUND CHANNEL
```

The script above was placed in the cast, and it flipped from one frame to the next one. When clicked, the button played the sound "Click". Note that puppetsound has to be reset (puppetsound 0) in order to keep it working.

```
on mouseUp
  puppetsound "Click"         --PLAY THE SOUND "CLICK"
  set TopOfMe to the locV of sprite 3 --SETS LOCAL VARIABLE
  TOPOFME TO THE VERTICAL COORDINATE OF THE MENU
  GRAPHIC, WHICH IS ITS UPPER LEFT CORNER

  if the mouseV < (TopOfMe+26) then --COMPARES THE POSITION
  OF THE CLICK IN INCREMENTS OF 26, WHICH IS THE LEADING OF
  THE MENU
    go frame "Bo1"
    puppetsound 0
  else
    if the mouseV > (TopOfMe+25) and the mouseV < (TopOfMe+52)
    then
      go frame "Bo2"
    else
```

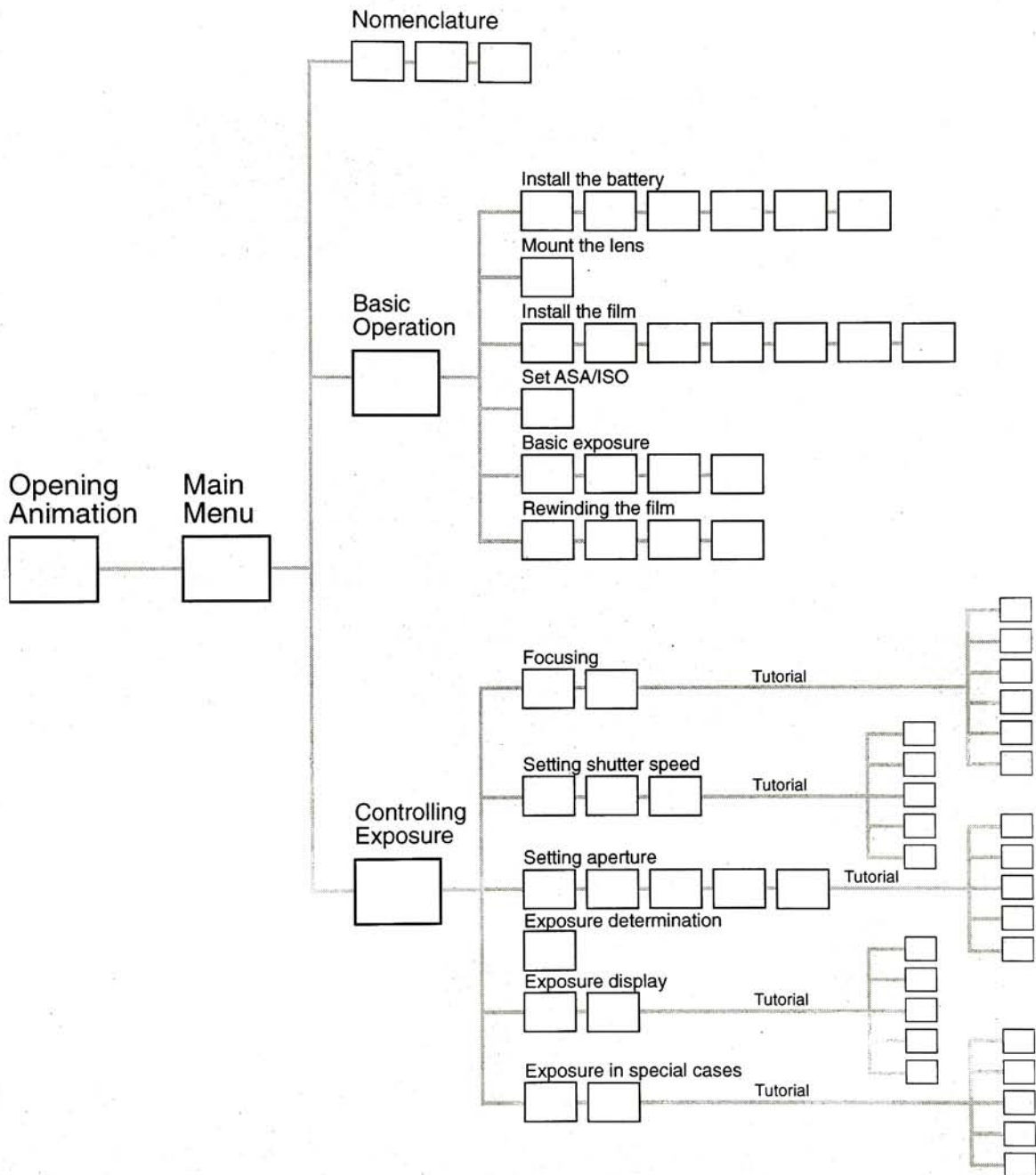


```

    if the mouseV > (TopOfMe+51) and the mouseV < (TopOfMe+78)
then
    go frame "Bo3"
    else
    if the mouseV > (TopOfMe+77) and the mouseV < (TopOfMe+104)
then
    go frame "Bo4"
    else
    if the mouseV > (TopOfMe+103) and the mouseV <
(TopOfMe+130) then
    go frame "Bo5"
    else
    go frame "Bo6"
    end if
    end if
    end if
    end if
    puppetSound 0
end mouseUp

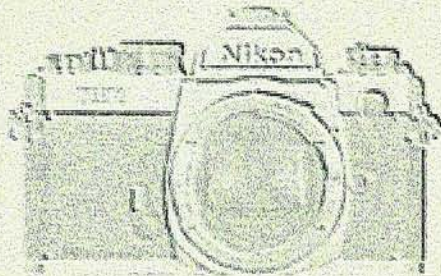
```

This script controlled the menus. When used, the script placed in the cast of the menus would look at which line was clicked, and execute an action defined under the condition "if." ("Bo5," "Bo6," etc., were marks placed in the score).



Basic structure of the Tutorial. The opening animation and the main menu can be reached from any point in the Tutorial.

Nikon FM2



A THESIS PROJECT BY
ROGER MARTINEZ i COLOM

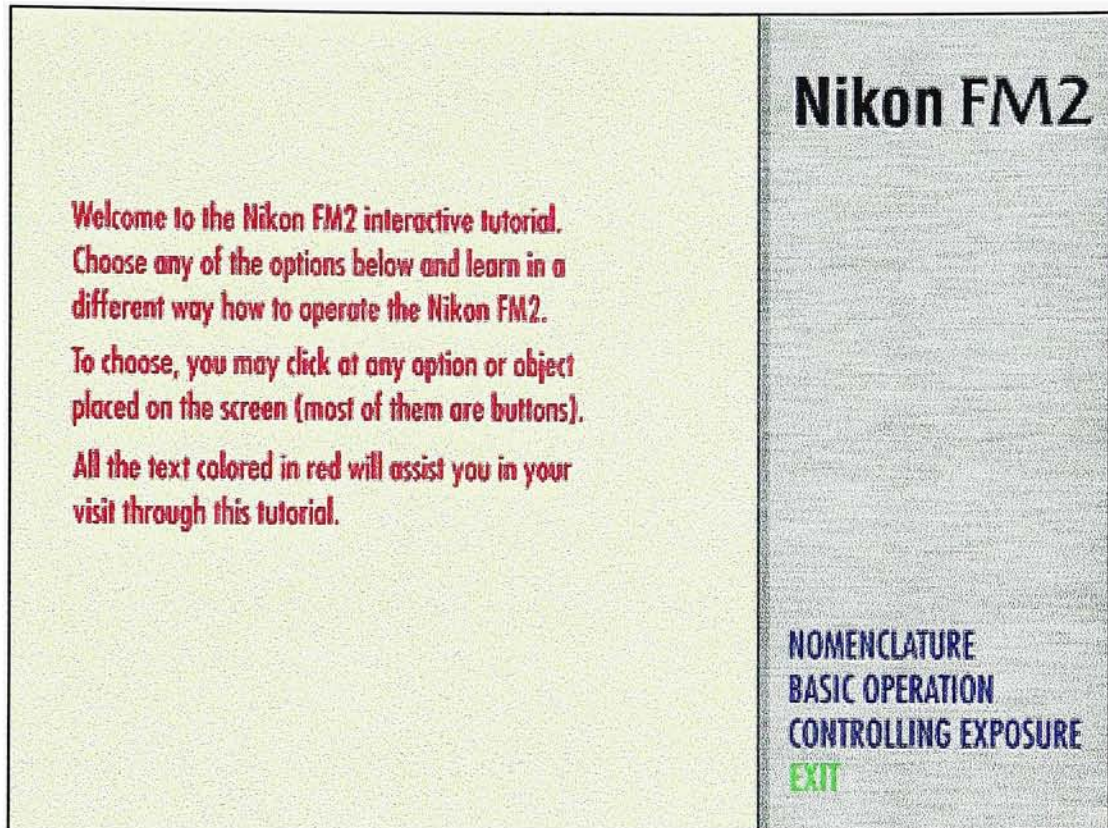
First, only the background is displayed. Then, sequentially, title, camera, and author appear by dissolving. The whole image (pictured here) remains on screen for a few seconds, and moves to the next screen.

Nikon FM2



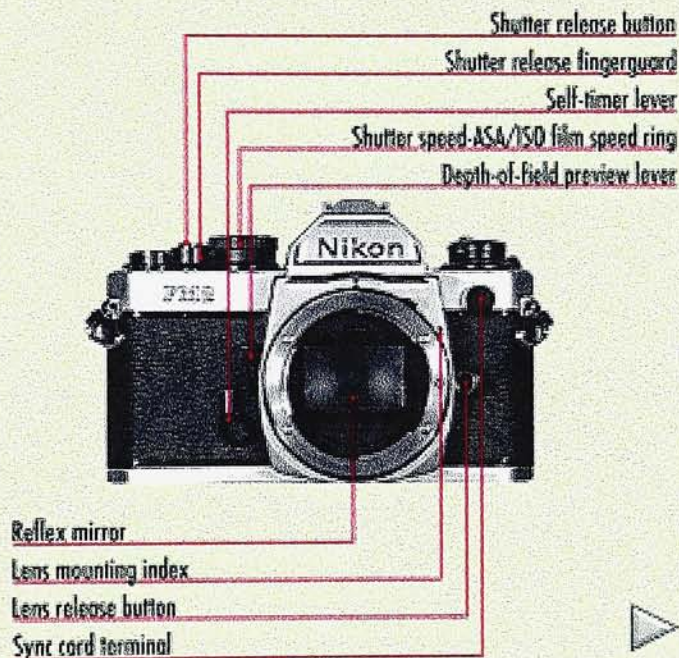
A THESIS PROJECT BY
ROGER MARTÍNEZ i COLOM

The image moves to the left of the screen, leaving space on the right for the menu panel, which will appear by dissolving.



The main menu appears on the right. On the left half, brief instructions help the user starting its voyage through the Tutorial.

NOMENCLATURE



Nikon FM2

Click at the arrows at the bottom of the screen to get more views of the Nikon FM2.

NOMENCLATURE
BASIC OPERATION
CONTROLLING EXPOSURE
EXIT

This is the first of the three screens that can be found under "Nomenclature". Different views of the camera and the name of its parts are shown on screen. The three different screens can be accessed by clicking at the bottom arrow.

BASIC OPERATION

This section will show you how to operate the FM2.
Click at the options on the right to go through the
basic steps on manipulating the camera.

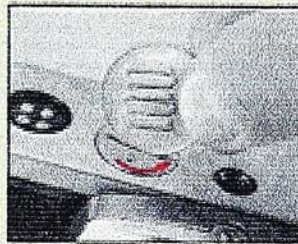
Nikon FM2

Install the battery
Mount the lens
Install the film
Set ASA/ISO
Basic Exposure
Rewinding the film

NOMENCLATURE
BASIC OPERATION
CONTROLLING EXPOSURE
EXIT

The main screen for the "Basic Operation" section. It also provides with basic instructions on how to operate. On the right side, submenus are shown in brown, and they are highlighted in red when selected.

BASIC OPERATION



1- Remove the battery chamber lid.

Use a coin to unscrew the lid in a counterclockwise direction.

Click the arrows to go to the next step...



Nikon FM2

Install the battery

Mount the lens

Install the film

Set ASA/ISO

Basic Exposure

Rewinding the film

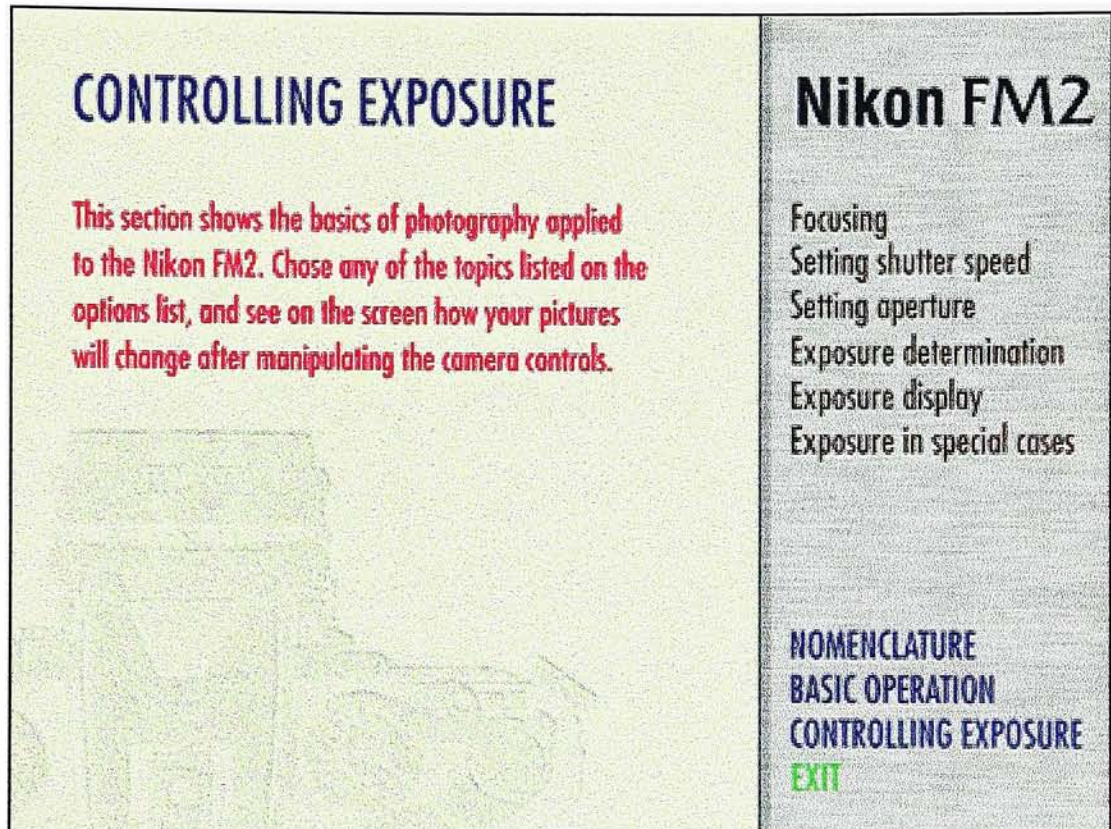
NOMENCLATURE

BASIC OPERATION

CONTROLLING EXPOSURE

EXIT

This is an example of what a submenu looks like. Different pictures with instructions show the user how to basically operate the camera. Again, the arrow at the bottom allows the user to navigate through the different screens.



We can see now the main screen for the section "Controlling Exposure". Brief instructions, just like before, are displayed to help the user understand what he can find in that section.

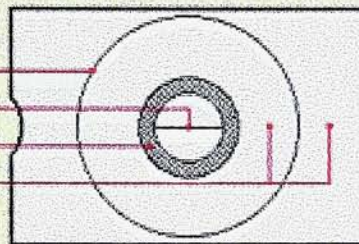
CONTROLLING EXPOSURE

12 mm-diameter reference circle

Split-image rangefinder spot

Microprism collar

Fine matte/Fresnel outer field



Viewfinder/viewing & focusing

The conventional way of securing correct focus is to turn the lens focusing ring until the image in the viewfinder appears sharp. The focusing system that comes with the camera as standard equipment has three focusing aids, shown above.

[Click here for the tutorial](#)



Nikon FM2

Focusing

Setting shutter speed

Setting aperture

Exposure determination

Exposure display

Exposure in special cases

NOMENCLATURE

BASIC OPERATION


CONTROLLING EXPOSURE

EXIT

Screen that shows how a subtopic screen looks like. Different concepts are displayed simultaneously with pictures and text. The arrow at the bottom flips through different screens, and it finally brings the user to the tutorial.

CONTROLLING EXPOSURE

Current focus:
0.6 m



Click at the arrows to turn the lens focusing ring, and observe the changes in the picture.

Nikon FM2

Focusing

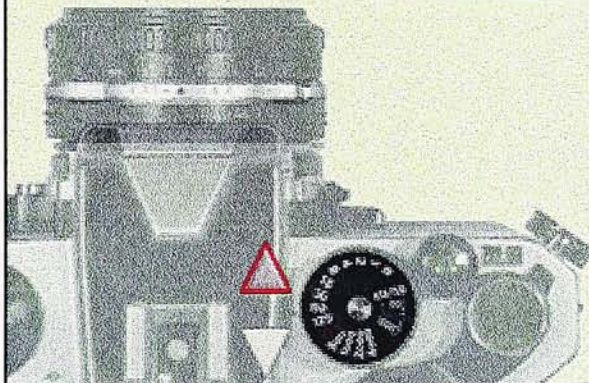
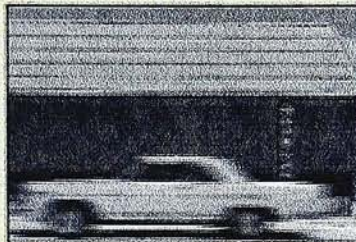
- Setting shutter speed
- Setting aperture
- Exposure determination
- Exposure display
- Exposure in special cases

NOMENCLATURE
BASIC OPERATION
CONTROLLING EXPOSURE
EXIT

First tutorial, on focusing. As the user clicks at the arrows next to the focusing ring, the image appears sharper and sharper, simulating the effect we observe in the viewfinder.

CONTROLLING EXPOSURE

Current speed:
1/60



Nikon FM2

Focusing

Setting shutter speed

Setting aperture

Exposure determination

Exposure display

Exposure in special cases

NOMENCLATURE

BASIC OPERATION

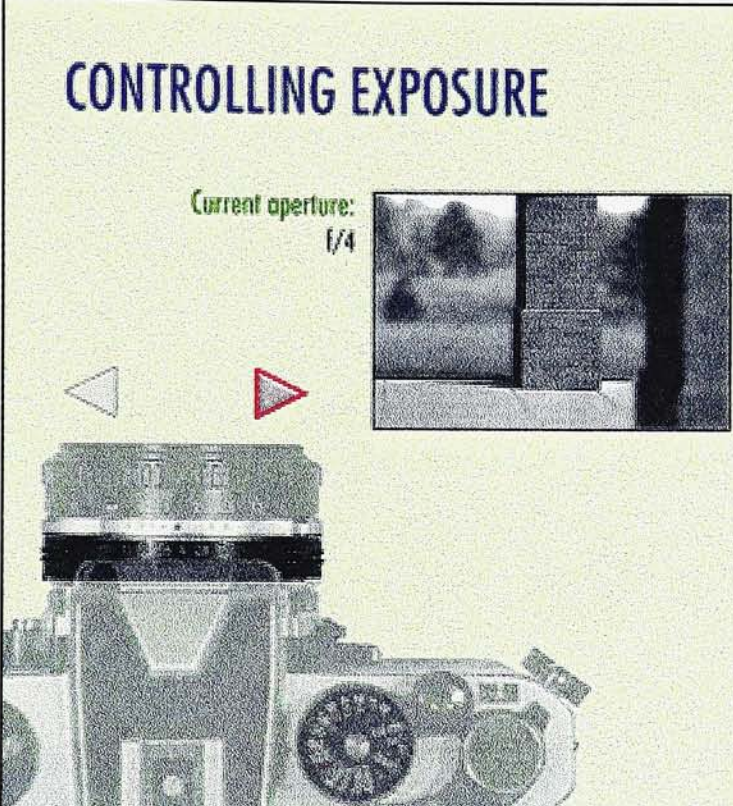
CONTROLLING EXPOSURE

EXIT

Second tutorial, on speed control. As the user turns the speed dial, the moving components of the image appear more blur, simulating the effect of speed and action on the print.

CONTROLLING EXPOSURE

Current aperture:
f/4



◀ ▶

Nikon FM2

- Focusing
- Setting shutter speed
- Setting aperture**
- Exposure determination
- Exposure display
- Exposure in special cases


NOMENCLATURE
BASIC OPERATION
CONTROLLING EXPOSURE
EXIT

Third tutorial, on aperture. When clicking the arrows next to the aperture ring, the image will increase or decrease the depth of field, simulating the real effect on the print.

CONTROLLING EXPOSURE

Current aperture:
f/16

Current speed:
1/60



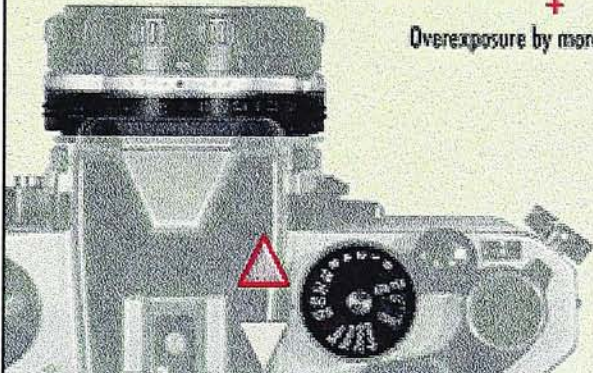
+

Overexposure by more than 1 stop

Nikon FM2

- Focusing
- Setting shutter speed
- Setting aperture
- Exposure determination
- Exposure display
- Exposure in special cases

- NOMENCLATURE
- BASIC OPERATION
- CONTROLLING EXPOSURE
- EXIT



Fourth tutorial, on exposure. As the user clicks at the arrows next to the speed dial, the exposure of the print will change, either over or underexposed.

CONTROLLING EXPOSURE

Click on different parts of the viewfinder to measure the exposure. Depending on where you measure, the final print will appear over or underexposed.



Nikon FM2

Focusing
Setting shutter speed
Setting aperture
Exposure determination
Exposure display
Exposure in special cases

NOMENCLATURE
BASIC OPERATION
CONTROLLING EXPOSURE
EXIT

The last tutorial, on special metering. Depending on where the user measures the light on the viewfinder (above), the print will look over or underexposed, simulating the zone system effects.

BIBLIOGRAPHY

Ansel Adams, The Negative (Toronto: Little, Brown and Company, 1992).

Barbara London Upton, John Upton, Photography (U.S.A.: Harper Collins, 1989).

Brenda Laurel (editor), The Art of Human-Computer Interface Design (U.S.A.: Addison Wesley Publishing Company, 1992).

Nikon Corporation, Nikon FM2: Instruction Manual (Japan: Nikon Corporation).

Time-Life Books, The Camera (New York: Time-Life Books, 1970).

Time-Life Books, The Print (New York: Time-Life Books, 1970).