

Praise for *Programming Cameras and Pan-Tilts with DirectX and Java*

This book will be an invaluable supplemental text for any image processing/computer vision course both at the undergraduate and graduate level. I would also consider it in lower level undergraduate programming courses! Many faculty members, including myself, with expertise in computer vision have started integrating computer vision research results into core courses like Data Structures and Algorithms. There will soon be strong industry demand for computer vision scientists and engineers—for people who understand vision technology and know how to apply it.

—Dr. George Bebis, Director, Computer Vision Lab,
University of Nevada

This book is an excellent source for the amateur programmer and the professional developer alike. It demonstrates the fundamentals for developing useful applications using both C++ with DirectX and Java with JMF, and it collects all of the critical information in one place.

—Jonathan Dowdall, University of Nevada, Reno

I commend the authors for the effort of providing tools and techniques for many users who are struggling to acquire an in-depth understanding of video programming. There is a growing need for this topic, but there exists limited useful information for the average user or developer who may not have enough knowledge in this particular field. Most documents are either not suitable for novice users or they are difficult to follow.

—Dr. Bir Bhanu, Professor of EECS and Director
Center for Research in Intelligent Systems,
University of California at Riverside



Programming Cameras and Pan-Tilts with DirectX and Java

Programming Cameras and Pan-Tilts with DirectX and Java

*Ioannis Pavlidis, University of Houston ■ Vassilios
Morellas, Honeywell, Inc. ■ Pete Roeber, Vital Images*



MORGAN KAUFMANN PUBLISHERS

AN IMPRINT OF ELSEVIER SCIENCE

AMSTERDAM BOSTON LONDON NEW YORK
OXFORD PARIS SAN DIEGO SAN FRANCISCO
SINGAPORE SYDNEY TOKYO

Senior Editor	Tim Cox
Publishing Services Manager	Edward Wade
Senior Production Editor	Cheri Palmer
Editorial Coordinator	Stacie Pierce
Project Management	Dusty Friedman, The Book Company
Cover Design	Yvo Riezebos Design
Cover Image	© National Geographic/gettyimages
Text Design	Lisa Devenish
Illustration/Composition	Interactive Composition Corporation
Copyeditor	Jane Loftus
Proofreader	Martha Ghent
Indexer	Micki Taylor
Printer	The Maple-Vail Book Manufacturing Group

Designations used by companies to distinguish their products are often claimed as trademarks or registered trademarks. In all instances in which Morgan Kaufmann Publishers is aware of a claim, the product names appear in initial capital or all capital letters. Readers, however, should contact the appropriate companies for more complete information regarding trademarks and registration.

Morgan Kaufmann Publishers
 An imprint of Elsevier Science
 340 Pine Street, Sixth Floor
 San Francisco, CA 94104-3205
www.mkp.com

© 2003 by Elsevier Science (USA)
 All rights reserved.
 Printed in the United States of America

07 06 05 04 03 5 4 3 2 1

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means—electronic, mechanical, photocopying, or otherwise—without the prior written permission of the publisher.

Library of Congress Control Number: 2002112509
ISBN: 1-55860-756-0

This book is printed on acid-free paper.

To our families

FOREWORD



This very timely book appears just as low-power, small, inexpensive silicon-based video cameras are becoming ubiquitous in our daily lives. Vast networks of distributed cameras across the Internet, as well as in intranets, enable high-bandwidth instantaneous communication either across the country or across different offices in a building. But how does all this data become integrated using the most modern computer technology? In most computer vision classes taught, there is a gap between the basic understanding of image-processing algorithm issues and how cameras are programmed at the system level to actually implement these algorithms. This new book by Pavlidis et al. bridges this gap with an excellent introduction and resource for how video cameras interface with computers with the most up-to-date protocols such as serial IEEE 1394 FireWire and USB. Furthermore, the most modern object-oriented programming paradigms, including Java, are used to show specific practical implementations including pan-tilt manipulation and object-tracking applications.

This book is recommended for motivated students or individuals who want to go beyond a typical laboratory computer vision undergraduate curriculum and get their hands dirty solving real-world problems by building working camera systems using PC programming for a host of important applications. A number of concrete examples are clearly presented, which can be immediately implemented and then built upon. This is an excellent supplement for most any experimental computer vision course, as well as for

computer engineers in commercial industry who are designing systems products requiring the serious use of camera technology.

Larry Wolff
Equinox Corporation
New York, New York

CONTENTS



FOREWORD ix

PREFACE xv

CHAPTER 1 **Introduction** 1

- 1.1 Ubiquitous Cameras and Computers—A Technological Revolution 2
- 1.2 Hardware Architecture 6
- 1.3 Software 13
- 1.4 Summary 13

CHAPTER 2 **The DirectShow Software Development Kit and the Component Object Model** 15

- 2.1 Overview 16
- 2.2 The COM 18
- 2.3 COM Interfaces 20
- 2.4 COM Objects 28
- 2.5 Our First DirectShow Application—Playing an AVI File 30
- 2.6 How to Write a DirectShow Filter 31
- 2.7 Summary 32

CHAPTER 3	A Video Preview Application—Source and Render Filters	33
3.1	The GraphEdit Utility	34
3.2	Building the Video Preview Filter Graph	34
3.3	The Property Pages of Filters	39
3.4	The AVI Decompressor	41
3.5	Summary	45
CHAPTER 4	A Custom Filter for Video Processing	47
4.1	A Simple Change Detection Filter	48
4.2	Define and Instantiate Our Filter's Class	50
4.3	Override the Appropriate Base Class Member Functions	58
4.4	Access Additional Interfaces	74
4.5	Create the Property Page	80
4.6	Create Registry Information	92
4.7	Summary	94
CHAPTER 5	A Simple Video-Based Security Application	97
5.1	Building the Application Shell	98
5.2	The Document-View Programming Architecture	102
5.3	The CCDAppApp Class	105
5.4	The CMainFrame Class	108
5.5	The CCDAppView Class	111
5.6	Summary	117
CHAPTER 6	A Custom Filter for Pan-Tilt Manipulation	119
6.1	Device Driver Categorization	120
6.2	Class CSerial: The Low-Level Serial Device Driver	122
6.3	Class CPanTiltDevice: The High-Level Driver	130

6.4	The CPanTiltFilter Source Filter	136
6.5	Running the CPanTiltFilter Using the GraphEdit Utility	147
6.6	Summary	149
CHAPTER 7	A Custom Filter for Object Tracking	151
7.1	The Theory of Mean-Shift Algorithm	152
7.2	A Practical Implementation of the Mean-Shift Algorithm	154
7.3	The Mean-Shift Tracking Algorithm	157
7.4	The Color Histogram CColorHistogramIndex Class	158
7.5	The CModelParameters Class	160
7.6	Implementation of the Mean-Shift Object-Tracking Filter	161
7.7	Summary	178
CHAPTER 8	An Object-Tracking Application	179
8.1	Introduction	180
8.2	Running the PTTracking Application	180
8.3	Main Control Dialog	184
8.4	PTTracking Application Window	186
8.5	Summary	206
CHAPTER 9	Computer-Camera Programming in Java	207
9.1	Understanding JMF	208
9.2	The Java Live Viewer Application—Video Acquisition and Display Using Java	208
9.3	The Java Video Filter Application—Pixel Manipulation Using Java	219
9.4	Video Data Transmission for the Web Using Real-Time Transfer Protocol	225
9.5	Summary	250

APPENDIX A	Component Software Setup	251
A.1	Visual Studio .NET Installation	251
A.2	DirectX 8.1 Software Development Kit Installation	251
A.3	Configuring the Application Environment	254
A.4	Camera Drivers	256
APPENDIX B	Accessing a Live Device in JMStudio	257
B.1	Live Devices in JMStudio	257
	BIBLIOGRAPHY	259
	INDEX	261

Video cameras, especially digital video cameras, that easily interface with PCs proliferate at a record pace. Cameras interface with PCs through devices that are widely known as frame grabbers. Frame grabbers are usually special Peripheral Component Interconnect (PCI) cards that are added on to the basic PC configuration. In recent years, with the advent of Universal Serial Bus (USB) and the IEEE 1394 standard, the future of traditional frame grabbers seems doubtful. Both USB and IEEE 1394 are universal communication protocols over serial expansion buses. They provide enhanced PC connectivity, not only with digital cameras, but also with a number of other peripheral devices. In this book, when we mention frame grabbers we refer to both the traditional cards and the new serial expansion buses. Irrespectively of the interfacing hardware, program control over the video frames usually depends on the Windows Software Development Kit (SDK) DirectShow. DirectShow function calls can be made from within a Visual C++ program. Alternatively, program control can be exercised by Java. The relative advantage of Java is easier Web connectivity. The proliferation of PC-based cameras has started to popularize an array of applications where computer video programming plays a crucial role. Such applications include smart-user interfaces (eye-controlled), videoconferencing, automated security and surveillance, industrial inspection, and video games.

One of the most intriguing possibilities when you connect a camera to a PC is the ability to program the device to view the video feed across the Internet and through a Web browser. This gave rise to a

new generation of small and cheap digital cameras for PCs that are aimed particularly for videoconferencing and remote monitoring. Web cameras usually conform to the USB connectivity standard. The programming methods explained in this book apply equally well to any camera that connects to a PC through a digital interface or a DirectShow-compliant frame grabber.

In this book we address the issue of computer video programming with Visual C++ and Java in the Windows XP environment. Our book describes step-by-step how to develop Visual Studio .NET and Java Media Framework programs that interface with frame grabbers. The emphasis of the book is on the programming aspect of video acquisition, display, and processing. We provide brief explanations, however, of hardware and software configuration issues where appropriate for completeness. We also provide clear explanations of relevant computer vision concepts when we introduce video-processing coding examples. Special attention is given to the development of Web-based video-processing applications through the use of Java.

This book comes at a very critical time. PC cameras and cameras in general are becoming ubiquitous and communicate with computers one way or another. This trend is fueling the demand for developers who are well versed in computer video-processing applications. Our book aspires to cover the present gap in the relevant literature. It targets primarily professional developers who work or are about to work in the area of computer video programming. The book can also be used as a complementary programming book in upper-level undergraduate or graduate level courses in image processing, computer vision, and multimedia. We assume that the prospective reader has a basic knowledge of the Visual C++ and Java environments.

Our book has a number of outstanding features:

- It emphasizes computer video programming and not simple image processing. The readers will be able to program real devices (cameras) for a number of emerging applications.
- It emphasizes practical working examples. We believe that learning by example is the best way for someone to learn programming and, in particular, to learn computer video programming. For each example we provide a clear and down-to-earth explanation of the theoretical aspects and a fully documented recital of the complete code.

- It includes such unique topics as programming of pan-tilt devices for computer control of camera motion.
- It is accompanied by a CD that contains all the code described in the book.

Acknowledgments

We would like to thank our families for their patience throughout the book-writing process. We would also like to thank our acquisitions editor Tim Cox and his staffer Stacie Pierce for being incredibly forgiving. We are equally thankful to Cheri Palmer and Dusty Friedman from the production team for a job well done. Finally, we are indebted to our reviewers for their valuable comments.

