Introduction to Computer Vision using OpenCV
Eric Gregori

Embedded Systems Conference, Boston, September 2012

Berkeley Design Technology, Inc.
Oakland, California USA
+1 (510) 451-1800

info@BDTI.com
http://www.BDTI.com

What is OpenCV?

An open source library of over 500 functions

Over 2 dozen examples

An easy tool for experimenting with computer vision

C/C++/Python API

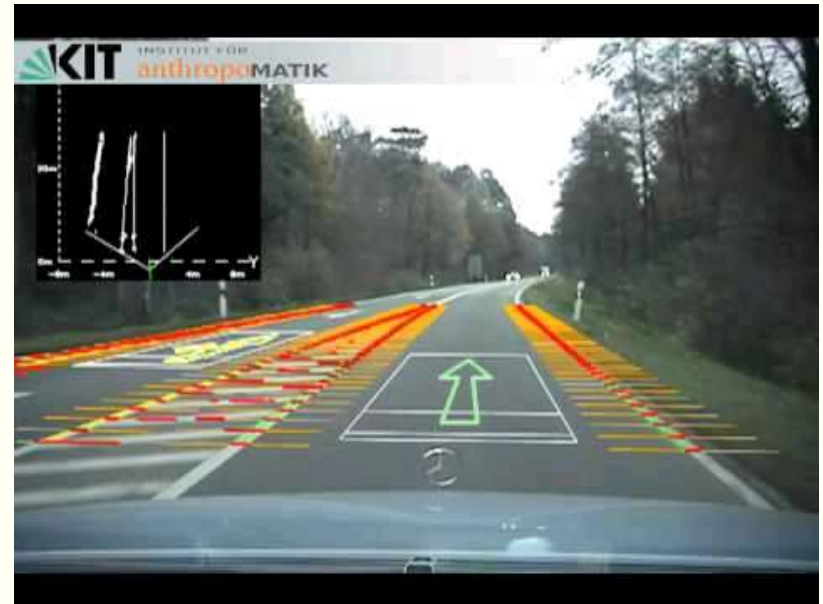Windows/Linux/ Android/iPhone platforms

Over 3,000,000 downloads

What Can OpenCV Do?

ANALYSIS • ADVICE • ENGINEERING
FOR EMBEDDED PROCESSING TECHNOLOGY

OpenCV in The Embedded Space

OpenCV has always been available to the embedded space under Linux.

The library has been ported to: PowerPC, MIPS, Blackfin, Xscale and ARM.

If it can run Linux, it can run OpenCV.

http://whatnicklife.blogspot.com/2010/05/beagle-has-2-eyes-opencv-stereo-on.html

On YouTube, you can find demonstrations of OpenCV running on the TI Beagleboard, Freescale i.MX53 Quick Start Board, and various NVIDIA based tablets and phones.

In the mobile market you can find examples of OpenCV running on the iPhone and of-course Android devices.

OpenCV Licensing

"OpenCV is released under a BSD license, it is free for both academic and commercial use."

"The BSD License allows proprietary use, and for the software released under the license to be incorporated into proprietary products. Works based on the material may be released under a proprietary license or as closed source software. This is the reason for widespread use of the BSD code in proprietary products, ranging from Juniper Networks routers to Mac OS X."

Try OpenCV for Yourself

- BDTI has created some interactive examples to demonstrate just a small part of what you can do with OpenCV.
- The build tools, OpenCV libraries, and examples are shipped as a VMware image.
- The BDTI OpenCV VMware image provides an easy to use pre-built environment to get you up and running on OpenCV in minutes.
- Simply download the free VMware player and BDTI OpenCV VMware image and get started in developing with OpenCV now.

- You can download the BDTI OpenCV VMware image from the Embedded Vision Alliance website at:
- http://embedded-vision.com/platinum-members/bdti/embedded-vision-training/downloads/pages/OpenCVVMwareImage
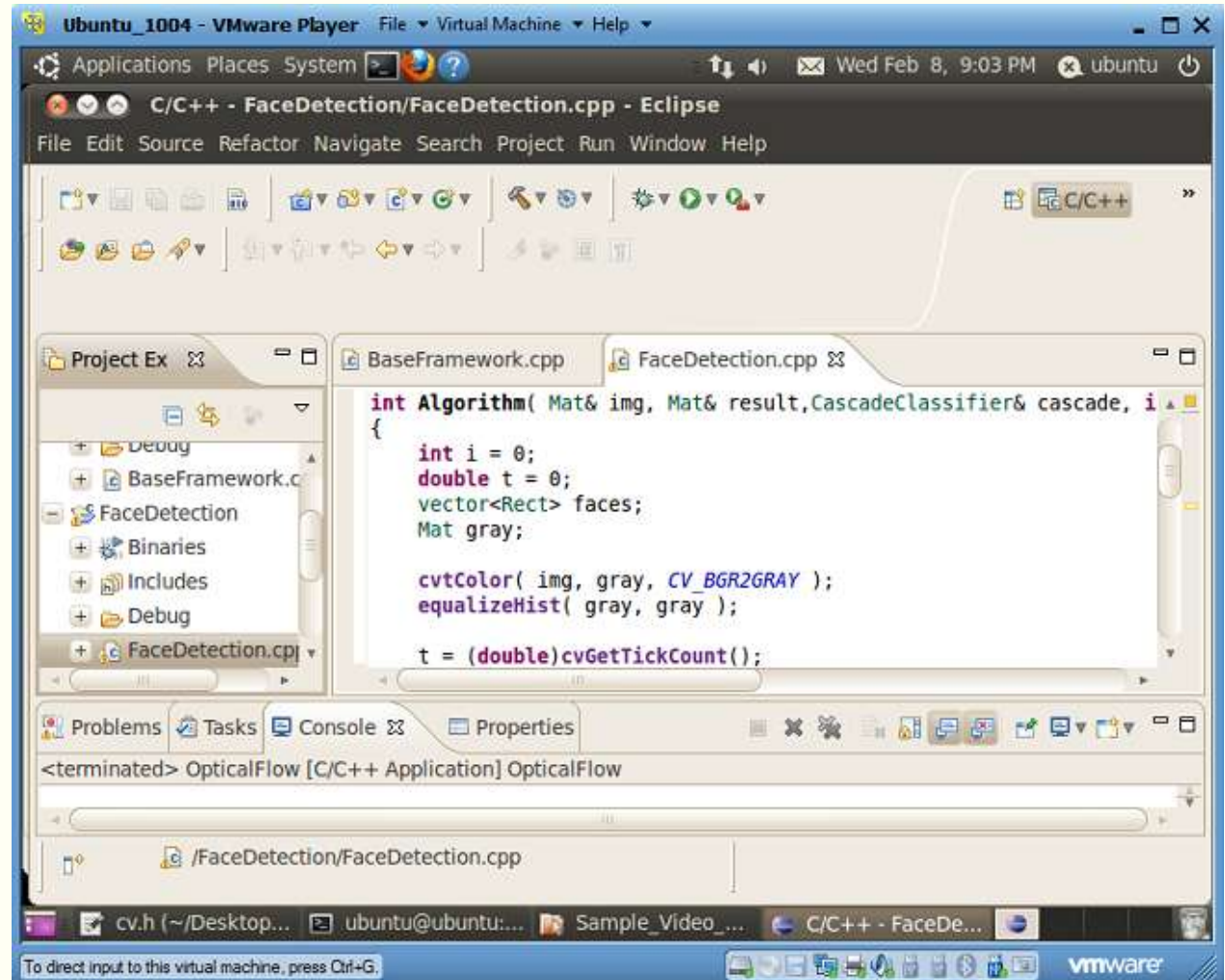
GETTING STARTED WITH OPENCV THE EASY WAY

All The Installation and Configuration Has Been Done for You

The Ubuntu OS and GCC compiler runs in a VMware image.

OpenCV is pre-installed and configured with all source.

Example applications use the Eclipse graphical debugging environment.

The Basics—An "Image" and A "Frame"

- Both images and frames are made up of individual pixels organized in a 2 dimensional array.

- For a color image, each pixel can be anything from 8 to 32 bits wide.

- Most monochrome images use 8 bits per pixel.

- A frame is a single image in a video sequence.

pixel

Y

X
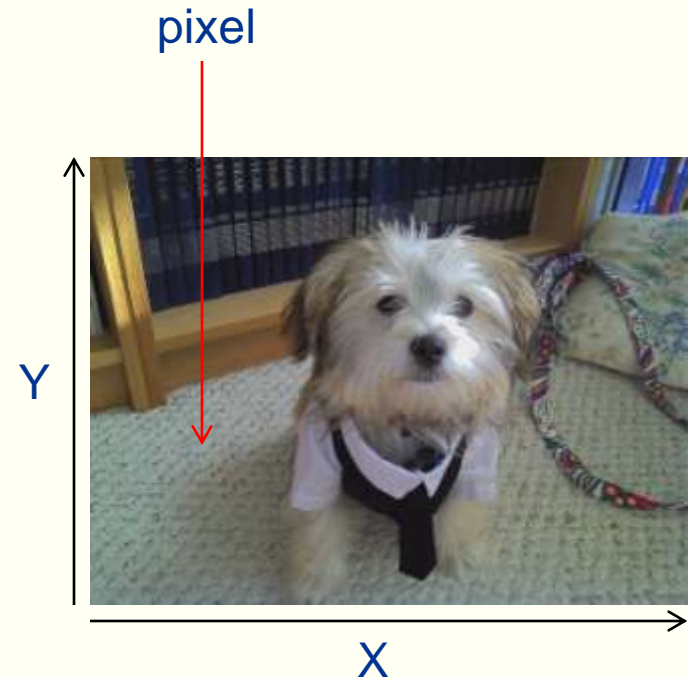
"Feature"—A Fundamental Concept in Computer Vision

feature *(fchr)n.*

A prominent or distinctive aspect, quality, or characteristic: a feature of one's personality; a feature of the landscape.
http://www.thefreedictionary.com/feature

The concept of, "a feature of an object" is very important for most computer vision algorithms.
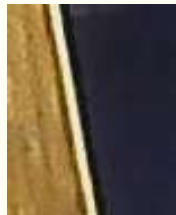
In a image or frame, **a feature is a _group of pixels_ with some _unique_ attribute.**

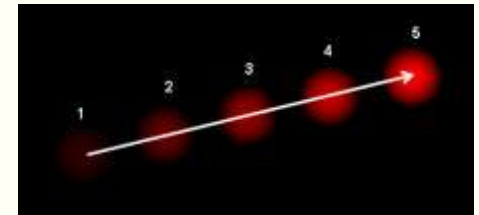corner   points   edge   contrast   motion

Some Basic "Building Block" Algorithms Used in Computer Vision

- **Detection**

    - Motion Detection—Finds groups of pixels (features) that are in motion (change in position from one frame to the next).

    - Line Detection—Finds groups of pixels (features) that are organized in straight lines, along edges.

    - Face Detection—Finds groups of pixels organized in a group that fits the template of a face.

- **Tracking**

    - Optical Flow based tracking—A combination of algorithms used to track moving objects in a video using features.
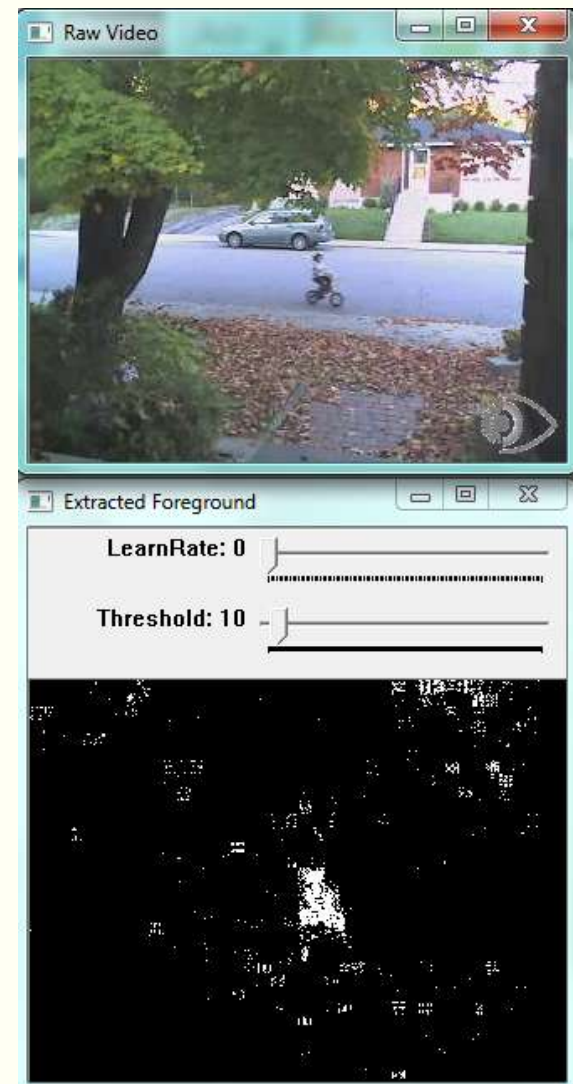
MOTION DETECTION

Motion Detection

- Motion detection in this context is done using frame subtraction, commonly referred to as background subtraction.

- The video is converted to monochrome and each pixel in the previous frame is subtracted from the current frame.

- If nothing changed between frames, the result of all the pixel subtractions will be 0.
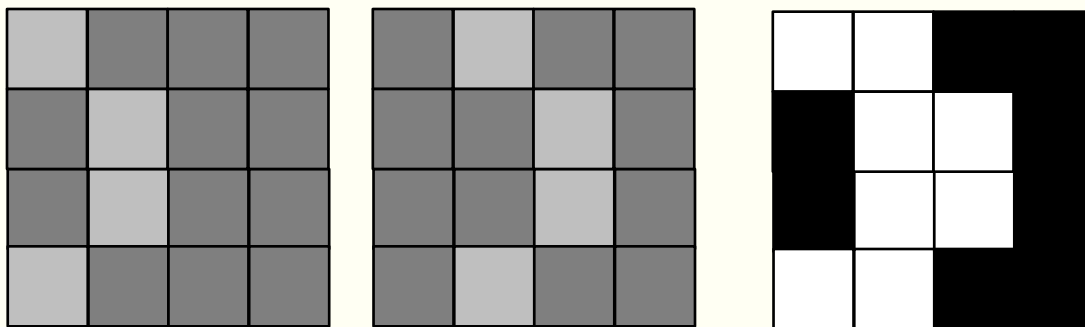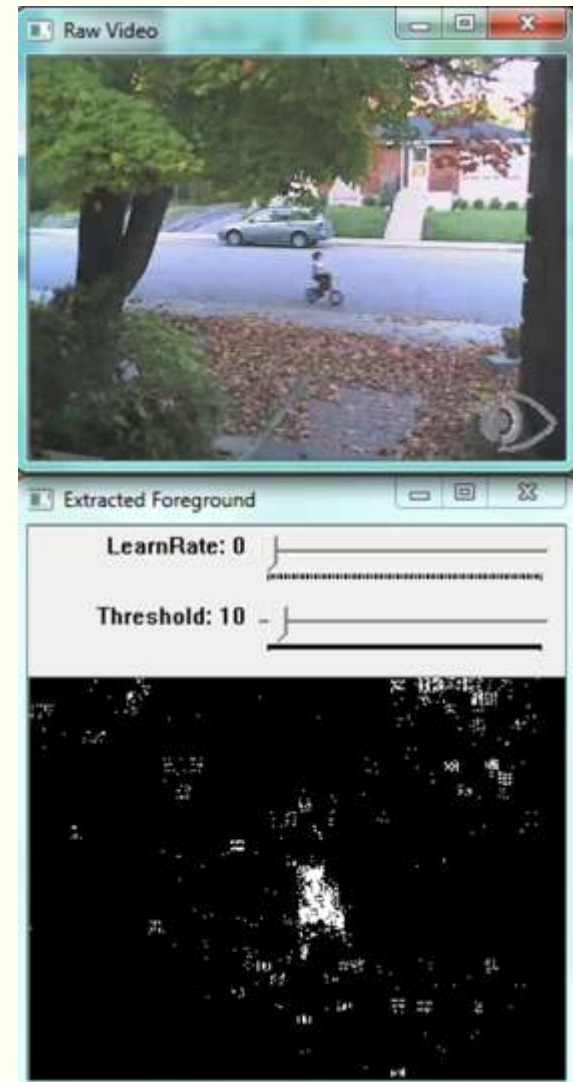
previous frame – current frame   =

Demo 1—Motion Detection

- **LearnRate**—Regulates the update speed (how fast the accumulator "forgets" about earlier images).

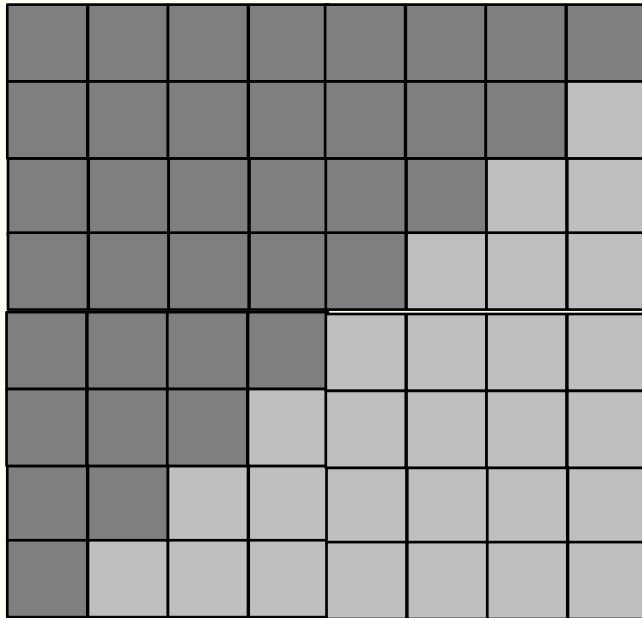- **Threshold**—The minimum value for a pixel difference to be considered moving.
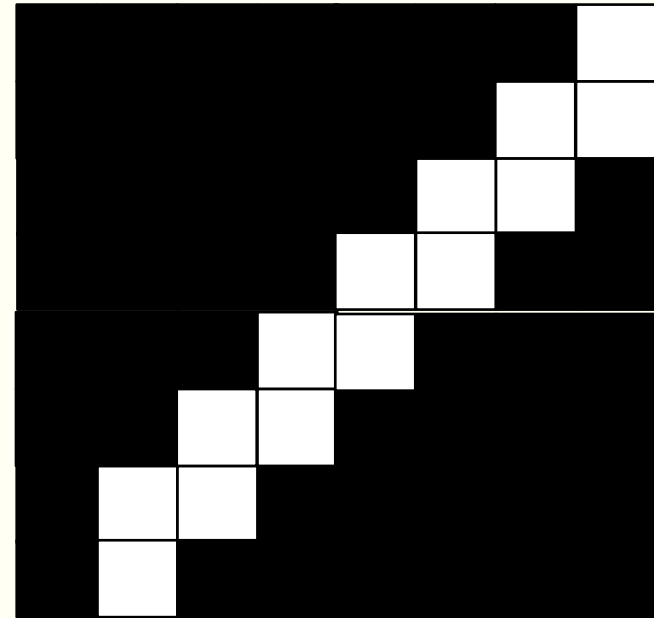
LINE DETECTION

Line Detection Using Edges

In a monochrome image, a line is defined as a group of pixels organized along a straight edge.

An edge in a monochrome image is defined as a dark pixel next to a lighter pixel.
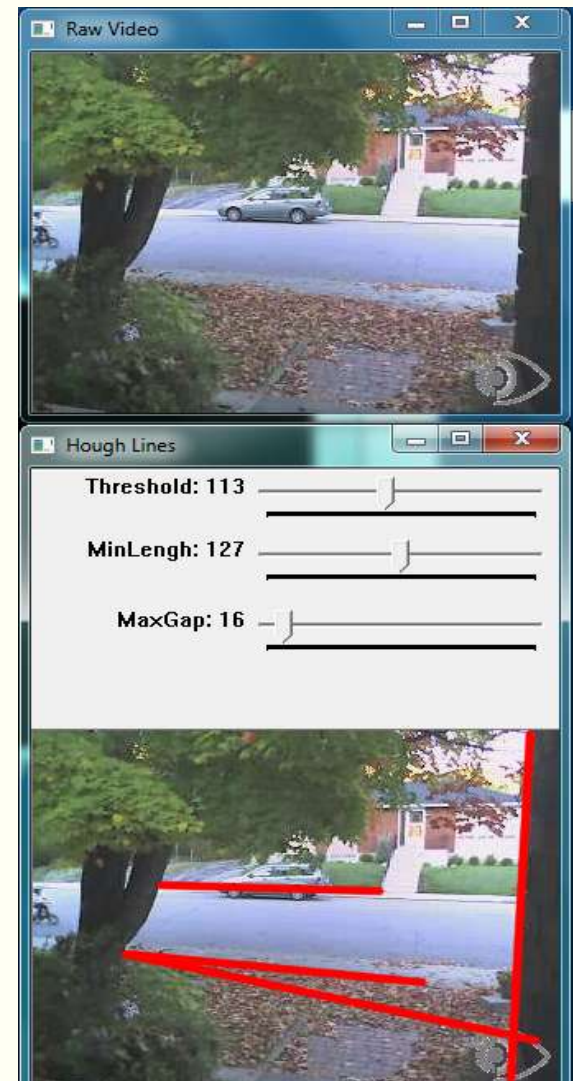
original image

detected edge

Line Detection Using Edges

- After the edge detector finds the edges, another algorithm called the Hough transform finds edge pixels that line-up in straight lines.

- Straight lines are a valuable feature in a image.

- Straight lines define the boundaries of objects in an image and can be used for tracking purposes in a video stream.

Demo 2—Line Detection Using Edges

- **Threshold—**Set the minimum difference between adjoining groups of pixels to be classified as an edge.

- **MinLength**—The minimum number of "continuous" edge pixels required to be classified as a straight line.

- **MaxGap**—The maximum number of missing edge pixels within a straight line, while still being considered "continuous".
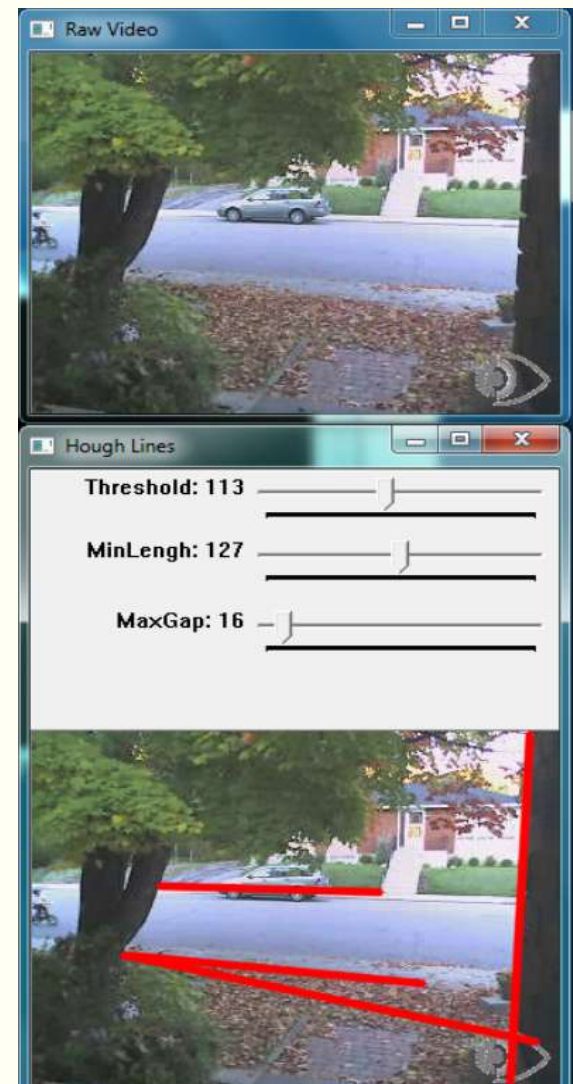
FACE DETECTION

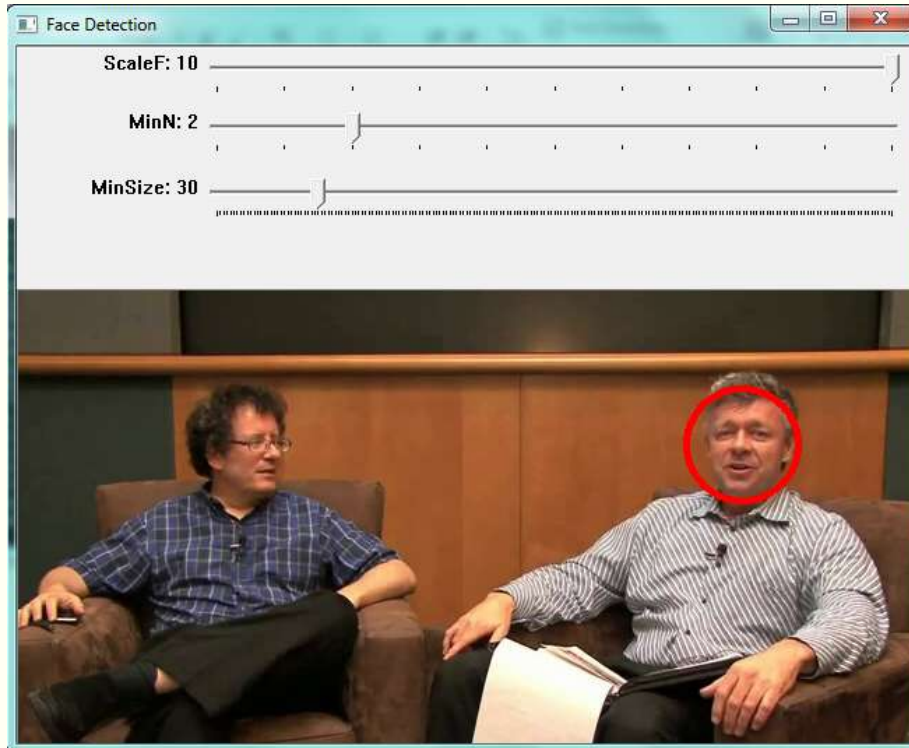# Face Detection

As the name implies, face detection is used to find faces in an image. The underlying algorithm can actually be used to detect any object. This algorithm is trained to look for specific features, in a specific order.

Training is done offline, and is accomplished by "showing" the learning algorithm both positive and negative images (images with a face and without a face).

The result of the training is a file that describes the object to detect, using very specific features.

Face Detection

Face detection, as demonstrated in this context, uses a set of **four** distinct templates to define **unique features**. Templates are used because they can be processed faster then other techniques. The template is laid over a portion of the image, and a weight is calculated based on the pixels under the template.
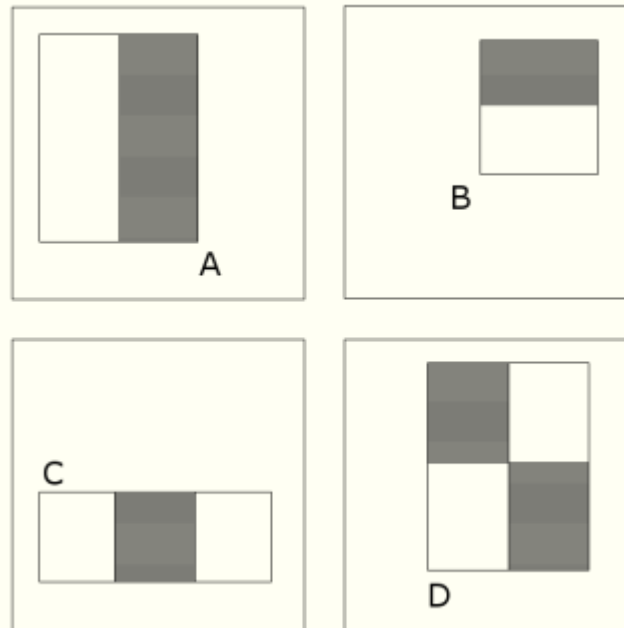
Face Detection

How does training work?

A face of 24 24 pixels can have 45,396 possible combinations/scales of the templates from the pervious slide.
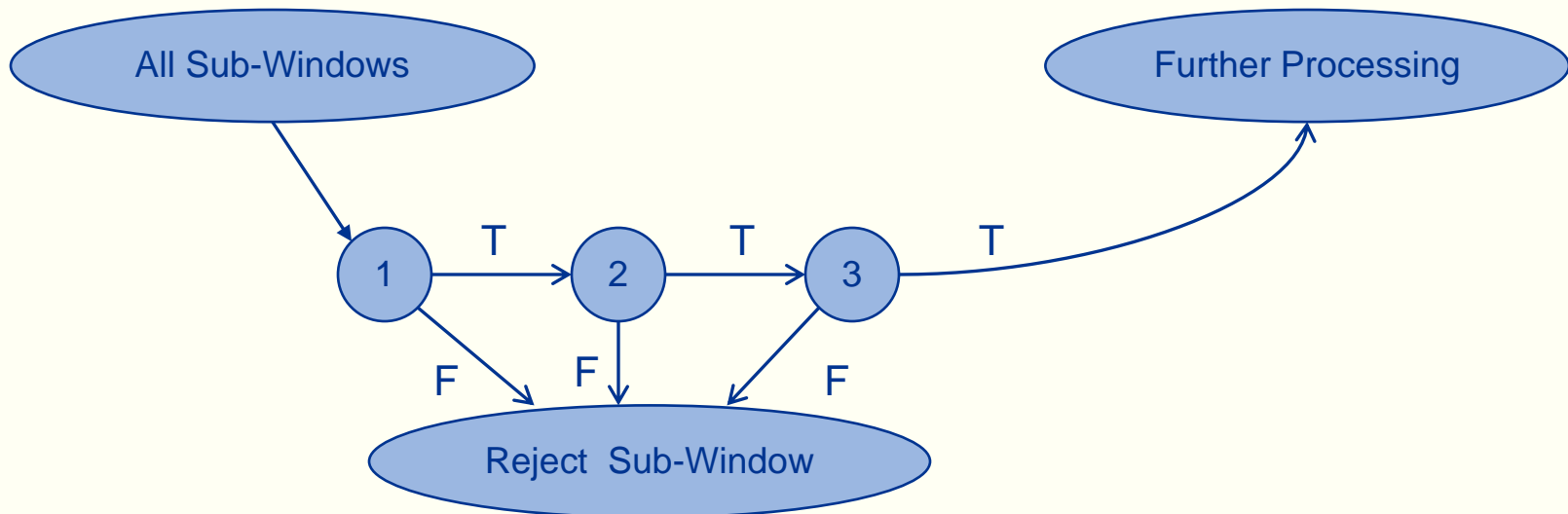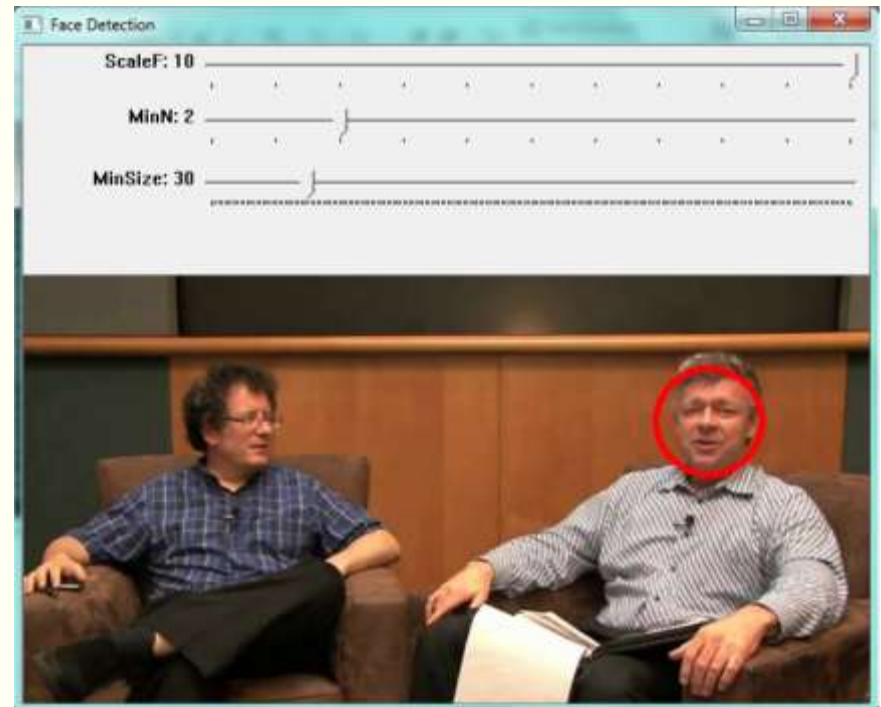
The purpose of training is to reduce the 45,396 possible combinations down to a minimum number and an ideal order.

Demo 3—Face Detection

- **MinSize**—The smallest face to detect. As a face gets further from the camera, it appears smaller. This parameter also defines the farthest a face can be from the camera and still get detected.

- **MinN**—The Minimum Neighbor parameter groups faces that are detected multiple times into one detection.

- **ScaleF**—Scale Factor determines the number of times the face detector is run at each pixel location. The Haar Cascade (xml file) that determines what the detector will detect, is designed for an object of only one size. In order to detect objects of various sizes (faces close to the camera as well as far from the camera) the detector must be scaled.

OPTICAL FLOW BASED TRACKING

ANALYSIS • ADVICE • ENGINEERING
FOR EMBEDDED PROCESSING TECHNOLOGY

Optical Flow Based Tracking

Optical flow is the change in position of a group of pixels (feature) from one image to the next.
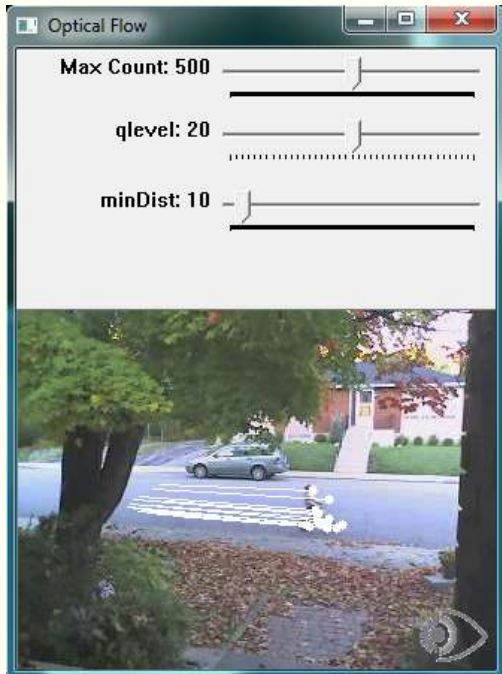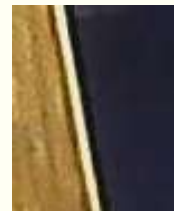
Optical Flow

The first step in optical flow is determining which features to use.

The algorithm used in this example uses features like corners, edges, and points of brightness.

These "good" features are found in each frame of a video stream.

With this data, a tracking algorithm can then be applied to predict where the object will appear in the next frame. This is object tracking.

Demo 4—Optical Flow Based Tracking

- **MaxCount** —The maximum number of good features to look for in a frame.

- **qlevel** —The quality of the features to accept. A higher quality feature is more likely to be unique, and be correctly found in the next frame. A low quality feature may get lost in the next frame, or worse be confused with another point in the image of the next frame.

- **minDist** —The minimum distance between features selected.
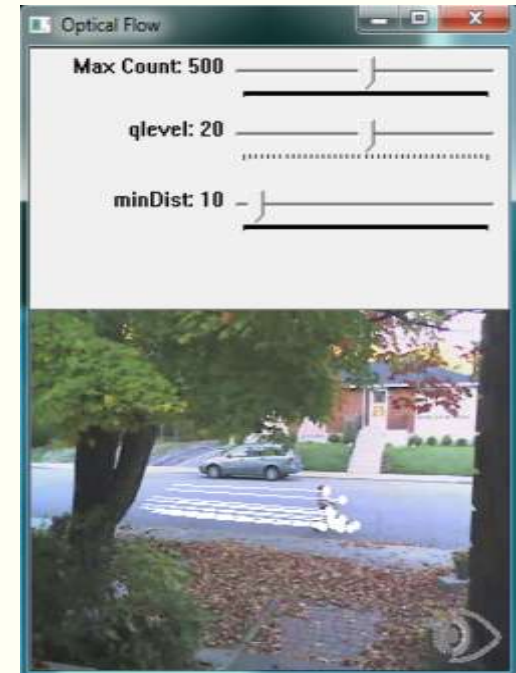
Summary

- Computer vision represents the "software sensor" of the future.

- Computer vision trades unique hardware for software.

- In some instances, computer vision can be considered a "software scalable sensor". As the available CPU horsepower increases, the capabilities of the technology increase.

- This class used OpenCV to demonstrate just a few algorithms available in the OpenCV library.

- OpenCV is a free computer vision library that has been downloaded over 3 million times.

- This presentation covered only 4 of the over 2,000 algorithms available in OpenCV.

To Probe Further

Visit the Embedded Vision Alliance web site at www.Embedded-Vision.com

BDTI provides consulting services to companies developing and using vision technology:

- Technology selection
- Product development engineering services
- Competitive analysis

Visit us at www.BDTI.com

RESOURCES

Embedded Vision Summit
A Free Educational Event for Engineers—Boston, September 19th

Learn how to use the coolest new technology in the industry to create "machines that see"

- Technical presentations on sensors, processors, tools, and design techniques
- Keynote by Prof. Rosalind Picard, MIT Media Lab
- Cool demonstrations and opportunities to meet with leading vision technology suppliers

Part of UBM Electronics' DESIGN East event

- DESIGN East also includes the Embedded Systems Conference, Sensors in Design, DesignMED, Android Summit, LED Summit, and exhibits

The Summit is free, but space is limited. To begin the registration process, send an email to summit@Embedded-Vision.com

Selected Resources: The Embedded Vision Alliance

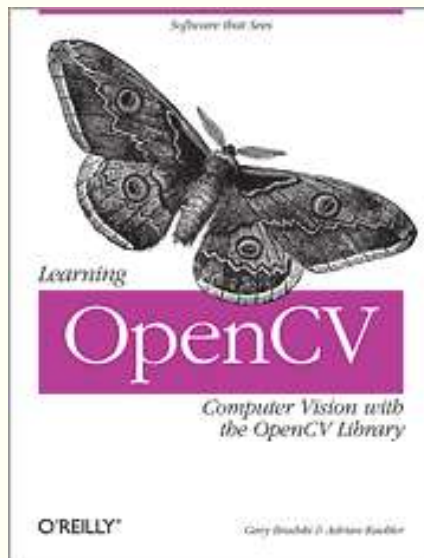The VMware image used in this presentation can be downloaded at: www.embeddedvisionacademy.com/vmwareimage.

Simply install the free VMware player and download the BDTI OpenCV VMware image from www.embeddedvisionacademy.com/vmwareimage.

Selected Resources

- OpenCV:
  - http://opencv.willowgarage.com/wiki/
  - Bradski and Kaehler, "Learning OpenCV: Computer Vision with the OpenCV Library", O'Reilly, 2008
  - Robert Laganière, "OpenCV 2 Computer Vision Application Programming Cookbook", Packt, 2011
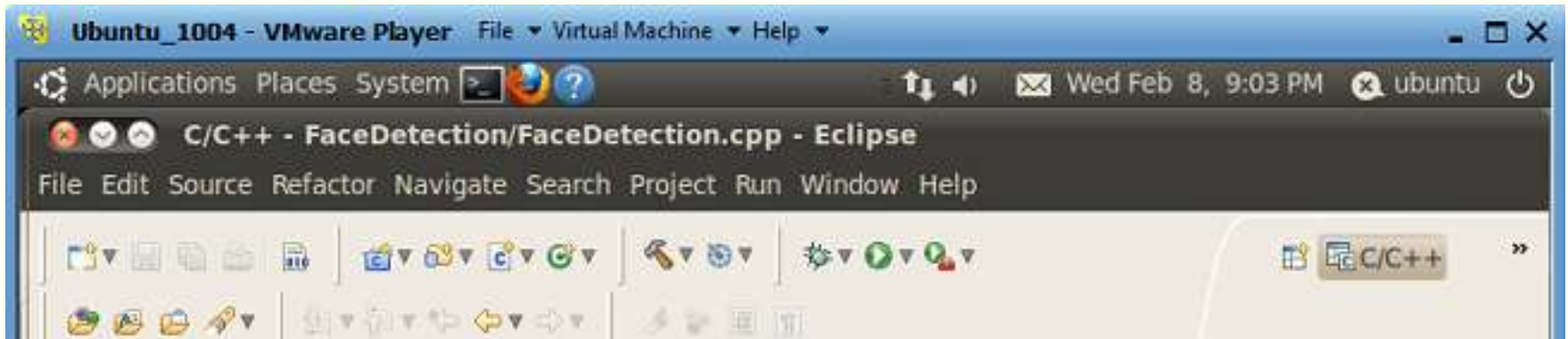
Selected Resources

Free VMware player: http://www.vmware.com/products/player/

Ubuntu 10.04 LTS: http://www.ubuntu.com/download/ubuntu/download

Eclipse CDT: http://www.eclipse.org/downloads/packages/eclipse-ide-cc-developers-includes-incubating-components/indigosr1

Additional Resources

BDTI's web site, www.BDTI.com, provides a variety of free information on processors used in vision applications.

BDTI's free "InsideDSP" email newsletter covers tools, chips, and other technologies for embedded vision and other DSP applications. Sign up at www.BDTI.com.

The "Embedded Vision Insights" newsletter showcases tutorials, interviews, and other videos, along with technical articles, industry analysis reports, news write-ups, and forum discussions that have recently appeared on the Embedded Vision Alliance's website. Sign up at http://www.embedded-vision.com/user/register.