Instructors:  - Prof Silvio Savarese (Stanford, CVGLab)
              - Dr Kari Pulli (Light)

CA:  - Saumitro Dasgupta  saumitro@stanford.edu
Agenda

- Administrative
  - Requirements
  - Grading policy

- Mobile Computer Vision

- Syllabus & Projects
Structure of the course

• First part:
  – Familiarize with android mobile platform
  – Work on two programming assignments on the mobile platform

• Second part:
  – Teams will work on a final project
  – Teams present in class 1-2 state-of-the-art papers on mobile computer vision
What you will learn

• How to program on the Android development platform

• State-of-the computer vision algorithms for mobile platforms

• Implement a working vision-based app on a mobile device
What you need to do

- Two programming assignments [40%]
- Course project [40%]
- Team presentations in class [15%]
- Class participation [5%]
- No midterms no finals!
Programming assignments [40%]

• Problem-0: warm up assignment (it won’t be graded)
  • Self-calibration; it will guide you through installing the Android platform and help you run a simple toy application

• Two problem sets [20% each]

• Each graded problem set includes
  – a programming assignment [15%]
  – a write-up [5%]

• Two topics:
  – Feature detection, descriptors, image matching, panorama and HDR image construction
  – Object and landmark recognition; Image classification
Programming assignments [40%]

• Programming assignments will be implemented on **Nvidia Shield**: a Tegra-based (K1) Android tablet
  – Simulators can only be used for debugging

• NVIDIA tablets are available for each student who is taking the course for credit.

• Tablets kindly lent by Nvidia

• Supporting material/tutorials will be based upon the Android platform
Programming assignments [40%]

- Important dates:
  - Problem-0
    - Released: 4/6 (next Monday)
    - Due: 4/13
  - First problem assignment:
    - Released on 4/13
    - Due on 4/24
  - Second problem assignment:
    - Released on 4/24
    - Due on 5/6
Course Project [40%]

• Goal: implement a computer vision application on a mobile platform
• We encourage students to use the NVIDIA Tegra-based Android tablet
• Students can use iOS for final project (but please let us know if this is the case)
  • NOTE: programming assignments must be completed on android

• Simulator can only be used for debugging
Course Project [40%]

• Teams: 1-3 people per team
  – The quality of your project will be judged regardless of the number of people on the team
  – Be nice to your partner: do you plan to drop the course?
Course Project [40%]

• Evaluation:
  • Proposal report 10%
  • Final report 20%
  • Final presentation 10%
Course Project [40%]

• Important dates:
  – Project proposal report due on 5/8
  – Project presentations on 5/27, 6/1, 6/3
  – Final project due: 6/7
Some examples of projects are:

• Recover the 3D layout of a room and augment it with new IKEA furniture
• Recognize your friend's face and link it to your friends on Facebook
• Localize yourself in a google map and visualize the closest restaurant on the smart phone's display
• Detect and face and turn it into a cartoon (and share it with friends)
• Create HDR panoramic images
• Recognize landmarks on the Stanford campus (e.g.: Memorial Church) and link it to relevant info from the web (Wikipedia, photos from other users, etc...)
Presentations in class [15%]

– Each student team will present 1-2 state-of-the-art papers on mobile computer vision
– Topics are pre-assigned but student teams can bid to present a paper of interest.
– Paper topics will be uploaded to the course syllabus soon.
– We are currently estimating a 15-30 minutes presentation per team

Evaluation:
  • Clarity of the presentation
  • Ability to master the topic
  • Ability to answer questions
Class participation [5%]

• Participate in class by attending, asking questions and participate in class discussions
  • During lectures
  • During team presentations

• In-class and piazza participation both count.

• Quantity and quality of your questions will be used for evaluating class participation.
Late policy

• No late submission for problem sets and Project

• Two “24-hours one-time late submission bonus" are available;
  • that is, you can use this bonus to submit your PA late after at most 24 hours. After you use your bonuses, you must submit on your assignment on time

• NOTE: 24-hours bonuses are not available for projects; project reports must be submitted in time.
Prerequisites

• CS131A, CS231A, CS232 or equivalent
• Familiar with C++ and JAVA
Collaboration policy

– Read the student code book, understand what is ‘collaboration’ and what is ‘academic infraction’.
– Discussing project assignment with each other is allowed, but coding must be done individually.
– Using on line presentation material (slides, etc…) is not allowed in general. Exceptions can be made and individual cases will be discussed with the instructor.
– On line software/code can be used but students must consult instructor beforehand. Failing to communicate this to the instructor will result to a penalty.
Agenda

• Administrative
  – Requirements
  – Grading policy

• Advanced topics in computer vision

• Syllabus & Projects
From the movie Minority Report, 2002
Computer vision and Mobile Applications

- More powerful machine vision
- Better clouds 😊
- Increase computational power
- More bandwidth
Panoramic Photography

AutoStitch

Panorama

Crop
Cancel

5580 x 3040
HDR

Intellsys
Digital photography

Auto-Correct.

1-tap automatic corrections adjust color, contrast, and brightness, so your photos shine.

Adobe Photoshop
3D modeling of landmarks
VLSAM

Project Tango

By Johnny Lee
Augmented reality
Visual search and landmarks recognition
Image search engines
Face detection
Computer vision and Mobile Applications

- EosSystems
- brickstream
- 2d3sensing
- digitalPersona
- Autostich
- MOBILEYE
- KINECT
- ebay
- Google Goggles
- kooaba
- Mirriad
- CREAFORM
- RUGMENT
- A9
- Photosynth

Timeline:
- 1990
- 2000
- 2010
Current state of computer vision

3D Reconstruction
- 3D shape recovery
- 3D scene reconstruction
- Camera localization
- Pose estimation

2D Recognition
- Image matching
- Object detection
- Texture classification
- Activity recognition

Mobile computer vision
Embedded systems

A special purpose computer system enclosed or encapsulated within a physical system

They are everywhere today!
- Consumer electronics
- Communication
- Entertainment
- Transportation
- Health
- Home appliances
Examples of embedded systems

- Video-assisted robots
- Medical imaging devices
Examples of embedded systems

- Video-assisted robots
- Medical imaging devices
- Autonomous cars
Examples of embedded systems

- Video-assisted robots
- Medical imaging devices
- Autonomous cars
- Smart phones
- Tablets
- Glasses
Smart phones & tablets: common characteristics

- Low cost
- Small package
- Resource constraints
- Real-time constraints (for some systems/applications)
Hardware Components

- Powerful mobile processors
- Dedicated chips for display driver, touchscreen control, GPS, bluetooth, WiFi more...
- Storage/Memory
- Display/touchscreen
- Communications/Connectivity
- Graphics
- Camera
- Accelerometer
- Compass/gyroscope
Powerful Mobile Processors

• Microcontrollers/Microprocessors—run OS, applications

• GPU – architected for fast rendering operations

• DSP – architected for fast, parallel vector operations

• Nvidia Tegra-K1 based Android tablet
Operating systems

Android OS

- Released in 2008
- Software platform based on Linux 2.6
- Developed by Google and the Open Handset Alliance
- Emulators on Mac, Windows, and Linux
- Version 5 (lollipop)
Challenges

• The mobile system has limited:
  computational power - bandwidth - memory
  - What to compute on the client (features, tracks)
  - How much data must be transferred to the back end
  - What to compute on the back end
  - How much data must be transferred back to the client to visualize results

• Computer vision algorithms with
  - Guaranteed (high)accuracy
  - Efficient (use little computational power/memory)
  - Fast (possibly real time)

• Many of these CV problems are still open
Client and Server paradigm

- Mobile system (client)
- Computing nodes (back end) (cloud, server)
- The internet; on-line repositories
In this class

• We will explore computer vision algorithms
  – Meet challenges above
  – Can be implemented on a mobile system
Class organization

Part 1:
• Review the Android architecture
• Introduce Android development platform

Part 2:
• Feature detection and descriptors
• Image matching
• Panorama and HDR images

Part 3:
• Object detection, landmark recognition and image classification
• Deep Learning and random forest

Part 4:
• Tracking, VSLAM, and virtual augmentation

Part 5:
• Special topics in mobile computer vision
• Project discussion and presentations
Class organization

Part 1:
- Review the Android architecture
- Introduce Android development platform

Part 2:
- Feature detection and descriptors
- Image matching
- Panorama and HDR images

Part 3:
- Object detection, landmark recognition and image classification
- Deep Learning and random forest

Part 4:
- Tracking, VSLAM, and virtual augmentation

Part 5:
- Special topics in mobile computer vision
- Project discussion and presentations
Features and descriptors

- Detectors:
  - DoG
  - Harris

- Descriptors:
  - SURF: Speeded Up Robust Features
  - Implementation on mobile platforms

[Bay et al 06]
[Chen et al 07]
Image matching

• M. Brown and D. Lowe, “recognizing panoramas”, 03
• Yingen Xiong and Kari Pulli, "Fast Panorama Stitching for High-Quality Panoramic Images on Mobile Phones", IEEE Transactions on consumer electronics, 2010
Automatic Panorama Stitching

- M. Brown and D. Lowe, “recognizing panoramas”, 03
- Yingen Xiong and Kari Pulli, "Fast Panorama Stitching for High-Quality Panoramic Images on Mobile Phones", IEEE Transactions on consumer electronics, 2010
Automatic Panorama Stitching

• M. Brown and D. Lowe, “recognizing panoramas”, 03
• Yingen Xiong and Kari Pulli, "Fast Panorama Stitching for High-Quality Panoramic Images on Mobile Phones", IEEE Transactions on consumer electronics, 2010
High Dynamic Range (HDR) images

Mertens, Kautz, van Reeth PG 2007
Class organization

Part 1:
• Review the Android architecture
• Introduce Android development platform

Part 2:
• Feature detection and descriptors
• Image matching
• Panorama and HDR images

Part 3:
• Object detection, landmark recognition and image classification
• Deep Learning and random forest

Part 4:
• Tracking, VSLAM, and virtual augmentation

Part 5:
• Special topics in mobile computer vision
• Project discussion and presentations
Visual search and landmarks recognition
Location/landmark recognition

Quack et al 08
Hays & Efros 08
Li et al 08

Courtesy of R. Szelisky and S. Seitz
Web repositories

Yeh et al 04
Zheng et al 09

(on the cloud; eg picasa)
Shape and object matching

- Shape Classification Using the Inner-Distance [Ling and Jacobs 07]
Bag of words representations

- Pyramid matching
- Recognition with a Vocabulary Tree

- K. Grauman and T. Darrell 2005
- S. Lazebnik et al, 2006
- D. Nister et al. 2006,

- Accuracy
- Efficiency
- Scalability to large database
Deep learning and random forests

Hinton, Bengio, Lecun, Ng, Breiman, Amit, Geman, etc....

Courtesy of Laura Skelton
Class organization

Part 1:
• Review the Android architecture
• Introduce Android development platform

Part 2:
• Feature detection and descriptors
• Image matching
• Panorama and HDR images

Part 3:
• Object detection, landmark recognition and image classification
• Deep Learning and random forest

Part 4:
• Tracking, VSLAM, and virtual augmentation

Part 5:
• Special topics in mobile computer vision
• Project discussion and presentations
Features from videos

- Features from videos
- Descriptors from videos
- On-line feature tracking

Courtesy of Jean-Yves Bouguet

- Ferrari et al 01
- Skrypnyk & Lowe 04
- Takacs et al 07
- Ta et al 09
- Klein & Murray 09
3D reconstruction and camera localization

- SFM
- VSLAM

• Ferrari et al 01
• Skrypnyk & Lowe 04
• Takacs et al 07
• Ta et al 09
• Klein & Murray 09

Courtesy of Jean-Yves Bouguet
Class organization

Part 1:
• Review the Android architecture
• Introduce Android development platform

Part 2:
• Feature detection and descriptors
• Image matching
• Panorama and HDR images

Part 3:
• Object detection, landmark recognition and image classification
• Deep Learning and random forest

Part 4:
• Tracking, VSLAM, and virtual augmentation

Part 5:
• Special topics in mobile computer vision
• Project discussion and presentations
Agenda

• Administrative
  – Requirements
  – Grading policy

• Mobile Computer Vision

• Syllabus
Next Lecture

• Overview of the Android platform & guiding examples by Dr Alejandro Troccoli (Nvidia)

• Please come and pick up your tablet!
  – Today until 6pm, 126 Gates
  – Tomorrow from 2-3pm, 126 Gates
Computer vision and Applications

3D

EosSystems
2d3 sensing
Microsoft Photosynth
CREAFORM

2D

Nikon
digitalPersona
TAAZ
Google Goggles
kooaba
A9
Image Recognition
Computer vision and Applications

New applications

EosSystems
2d3sensing
CREAFORM
Photosynth
AUGMENT

3D

Nikon
digitalPersona.
TAAZ
Google Goggles
kooaba

2D
How to make this to work?

Solve a number of challenging computer vision problems and implement them on a mobile system

- Image classification
- Object detection
- Tracking
- Matching
- 3D reconstruction
Detection & object recognition

Does this image contain a car? [where?]
Detection:
Which object does this image contain? [where?]
Detection & object recognition

Does this image contain a clock? [where?]
~10,000 to 30,000
Challenges: illumination

image credit: J. Koenderink
Challenges: scale
Challenges: deformation
Challenges: occlusion

Magritte, 1957
Challenges: background clutter

Kilmeny Niland. 1995
Challenges: viewpoint variation

Michelangelo 1475-1564

slide credit: Fei-Fei, Fergus & Torralba
Challenges: intra-class variation
Recognition

– Search strategy: Sliding Windows
  • Simple
  • Computational complexity \((x,y, S, \theta, N \text{ of classes})\)
    - BSW by Lampert et al 08
    - Also, Alexe, et al 10

Recognition

– Recognition task

– Search strategy: Sliding Windows

  • Simple
  
  • Computational complexity \((x, y, S, \theta, N \text{ of classes})\)
    
    - BSW by Lampert et al 08
    - Also, Alexe, et al 10

  • Localization
    
    • Objects are not boxes

Viola, Jones 2001,
Recognition

- Recognition task

- Search strategy: Sliding Windows
  - Simple
  - Computational complexity \((x, y, S, \theta, N \text{ of classes})\)
    - BSW by Lampert et al 08
    - Also, Alexe, et al 10
  - Localization
    - Objects are not boxes
    - Prone to false positive
      - Non max suppression: Canny ’86
      - Desai et al, 2009

Viola, Jones 2001,
Star models by Latent SVM

Felzenszwalb, McAllester, Ramanan, 08
• Source code:
Implicit shape models

- Visual codebook is used to index votes for object position

B. Leibe, A. Leonardis, and B. Schiele, *Combined Object Categorization and Segmentation with an Implicit Shape Model*, ECCV Workshop on Statistical Learning in Computer Vision 2004

Credit slide: S. Lazebnik
Face Recognition

- Digital photography
- Automatic face tagging
The Viola/Jones Face Detector


- A “paradigmatic” method for real-time object detection
- Training is slow, but detection is very fast
- Extensions to mobile applications
Single 3D Object Recognition

- No intra-class variation
- High view point changes
Single 3D Object Recognition

- Handle severe occlusions
- Fast!

Lowe. ’99, ’04
Hsiao et al CVPR 10
Single 3D Object Recognition

• Recognizing landmarks
Tracking and 3D modeling

G. Klein and D. Murray. Improving the agility of keyframe-based SLAM. In ECCV08, 2008.

Min Sun, Gary Bradski, Bing-xin Xu, Silvio Savarese, Depth-Encoded Hough Voting for Joint Object Detection and Shape Recovery, ECCV 2009
Min Sun, Gary Bradski, Bing-xin Xu, Silvio Savarese, Depth-Encoded Hough Voting for Joint Object Detection and Shape Recovery, ECCV 2009
Automatic Panorama Stitching

Sources: M. Brown
Automatic Panorama Stitching

• M. Brown and D. Lowe, “recognizing panoramas”, 03
• Yijing Xiong and Kari Pulli, "Fast Panorama Stitching for High-Quality Panoramic Images on Mobile Phones", IEEE Transactions on consumer electronics, 2010
Agenda

• Administrative
  – Requirements
  – Grading policy

• Advanced Topics in Mobile Computer Vision

• Syllabus & Projects