CS231M · Mobile Computer Vision
Spring 2014

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

CAs:
- Saumitro Dasgupta  saumitro@stanford.edu
- Francois Chaubard  fchaubar@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 one state-of-the-art paper 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%]

• Two problem sets [20% each]

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

• Two topics:
  – Feature detection, descriptors, image matching, panorama and HDR image construction
  – Feature tracking and matching; Camera localization, mapping and 3D scene reconstruction

• Skeleton code for each PA will be available with links to library functions; students will need to fill out code and make code running
Programming assignments [40%]

• Programming assignments will be implemented on an NVIDIA Tegra-based 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 donated by Nvidia

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

• Important dates:

  – First problem assignment is released on 4/16 and due on 4/23

  – Second problem assignment is released on 4/23 and due on 5/7
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 & presentation 10%
  • Final report 20%
  • Final presentation 10%
Course Project [40%]

• Important dates:
  – Project proposal report + class presentation on 5/12
  – Project presentations on 6/2 and 6/4
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 one state-of-the-art paper 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 20-25 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

- Better clouds 😊
- Increase computational power
- More bandwidth
Panoramic Photography
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

By Johnny Lee
Augmented reality
Fingerprint biometrics
Visual search and landmarks recognition
Image search engines
Face detection
Computer vision and Mobile Applications
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
Current state of computer vision

3D Reconstruction

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

Lucas & Kanade, 81
Chen & Medioni, 92
Debevec et al., 96
Levoy & Hanrahan, 96
Fitzgibbon & Zisserman, 98
Triggs et al., 99
Pollefeys et al., 99
Kutulakos & Seitz, 99
Levoy et al., 00
Hartley & Zisserman, 00
Dellaert et al., 00
Rusinkiewic et al., 02
Nistér, 04
Brown & Lowe, 04
Schindler et al., 04
Lourakis & Argyros, 04
Colombo et al., 05
Golparvar-Fard, et al. JAEI 10
Pandey et al. IFAC, 2010
Pandey et al. ICRA 2011
Savarese et al. IJCV 05
Savarese et al. IJCV 06
Microsoft’s PhotoSynth
Snavely et al., 06-08
Schindler et al., 08
Agarwal et al., 09
Frahm et al., 10
3D Reconstruction

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

Current state of computer vision

Levoy et al., 00
Hartley & Zisserman, 00
Dellaert et al., 00
Rusinkiewic et al., 02
Nistér, 04
Brown & Lowe, 04
Schindler et al, 04
Lourakis & Argyros, 04
Colombo et al. 05
Golparvar-Fard, et al. JAEI 10
Pandey et al. IFAC, 2010
Pandey et al. ICRA 2011
Savarese et al. IJCV 05
Savarese et al. IJCV 06
Microsoft’s PhotoSynth
Snavely et al., 06-08
Schindler et al., 08
Agarwal et al., 09
Frahm et al., 10
Current state of computer vision

2D Recognition

- Image matching
- Object detection
- Texture classification
- Activity recognition

Turk & Pentland, 91
Poggio et al., 93
Belhumeur et al., 97
LeCun et al. 98
Amit and Geman, 99
Shi & Malik, 00
Viola & Jones, 00
Felzenszwalb & Huttenlocher 00
Belongie & Malik, 02
Ullman et al. 02
Argawal & Roth, 02
Ramanan & Forsyth, 03
Weber et al., 00
Vidal-Naquet & Ullman 02
Fergus et al., 03
Torralba et al., 03
Vogel & Schiele, 03
Barnard et al., 03
Fei-Fei et al., 04
Kumar & Hebert ’04
He et al. 06
Gould et al. 08
Maire et al. 08
Felzenszwalb et al., 08
Kohli et al. 09
L.-J. Li et al. 09
Ladicky et al. 10,11
Gonfaus et al. 10
Farhadi et al., 09
Lampert et al., 09
Current state of computer vision

2D Recognition

- Image matching
- Object detection
- Texture classification
- Activity recognition

Turk & Pentland, 91
Poggio et al., 93
Belhumeur et al., 97
LeCun et al. 98
Amit and Geman, 99
Shi & Malik, 00
Viola & Jones, 00
Felzenszwalb & Huttenlocher 00
Belongie & Malik, 02
Ullman et al. 02
Argawal & Roth, 02
Ramanan & Forsyth, 03
Weber et al., 00
Vidal-Naquet & Ullman 02
Fergus et al., 03
Torralba et al., 03
Vogel & Schiele, 03
Barnard et al., 03
Fei-Fei et al., 04
Kumar & Hebert ’04
He et al. 06
Gould et al. 08
Maire et al. 08
Felzenszwalb et al., 08
Kohli et al. 09
L.-J. Li et al. 09
Ladicky et al. 10,11
Gonfaus et al. 10
Farhadi et al., 09
Lampert et al., 09
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
Why is this challenging?

• Many of these CV problems are still open

• The mobile system has limited:
  – Computational power
  – Memory
  – Bandwidth
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
Mobile systems

Rich functionalities

- Voice call
- Texting
- Surfing internet
- Music playback
- Map
- Email
- Weather
- Camera/camcorder
- Streaming video
- Gaming console
- Wireless modem
- ....
Smart phones: 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
Powerful Mobile Processors

• Microcontrollers/Microprocessors—run OS, applications
• GPU—architected for fast rendering operations
• DSP—architected for fast, parallel vector operations
• Nvidia Tegra-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
Other specs of the Android phone

- Network I/O: 3G and 802.11
- Camera: 5MP
- Sensors:
  - Accelerometer
  - Ambient Light
  - Proximity Sensor
  - Compass
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
  - visualize results

• Computer vision algorithms with
  - guaranteed (high)accuracy
  - efficient (use little computational power/memory)
  - fast (possibly real time)
Client and Server paradigm

- mobile system (client)
- The internet; on-line repositories
- Computing nodes (back end) (cloud, server)
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:
- Feature tracking
- 3D reconstruction and camera localization

Part 4:
- 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:
• Feature tracking
• 3D reconstruction and camera localization

Part 4:
• 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

Sources: M. Brown
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

LDR images

Weight maps
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:
- Feature tracking
- 3D reconstruction and camera localization

Part 4:
- Special topics in mobile computer vision
- Project discussion and presentations
Features from videos

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

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

Courtesy of Jean-Yves Bouguet
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
Tracking and Virtual Reality insertions

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:
• Feature tracking
• 3D reconstruction and camera localization

Part 4:
• Special topics in mobile computer vision
• Project discussion and presentations
Bag of words model

Image representation

- Pyramid matching
- Recognition with a Vocabulary Tree

- Accuracy
- Efficiency
- Scalability to large database

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

Yeh et al 04
Zheng et al 09

(on the cloud; eg picasa)
Visual search and landmarks recognition

Quack et al 08
Hays & Efros 08
Li et al 08
Location/landmark recognition

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

Courtesy of R. Szelisky and S. Seitz
Shape and object matching

- Shape Classification Using the Inner-Distance [Ling and Jacobs 07]
Shape matching

• Match shape against database
• Retrieve relevant information

Agenda

• Administrative
  – Requirements
  – Grading policy

• Mobile Computer Vision

• Syllabus
Next Lecture

• Overview of the Android platform. Guiding examples

• Please come and pick up your tablet!
Computer vision and Applications

EosSystems

2d3 sensing

CREAFORM

3D

Photosynth

AUGMENT

2D

Google Goggles

digitalPersona.

Nikon

TAAZ

A9

kooaba

IMAGE RECOGNITION
Computer vision and Applications

EosSystems
2d3 sensing
CREAFORM
Microsoft Photosynth
AUGMENT

3D

New applications

2D

Nikon
digitalPersona.
TAAZ
Google Goggles
kooaba

81
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?]
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, θ, N 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
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

+ GPS
Tracking and 3D modeling

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

G. Klein and D. Murray. Parallel tracking and mapping on a camera phone. In ISMAR’09, 2009
Detection and 3d Modeling

Min Sun, Gary Bradski, Bing-xin Xu, Silvio Savarese, Depth-Encoded Hough Voting for Joint Object Detection and Shape Recovery, ECCV 2009
Detection and 3d Modeling

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

- 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

Sources: M. Brown
Agenda

• Administrative
  – Requirements
  – Grading policy

• Advanced Topics in Mobile Computer Vision

• Syllabus & Projects