Univerza *v Ljubljani*

# Advanced CV methods
# Introduction

Matej Kristan

Laboratorij za Umetne Vizualne Spoznavne Sisteme,
Fakulteta za računalništvo in informatiko,
Univerza v Ljubljani

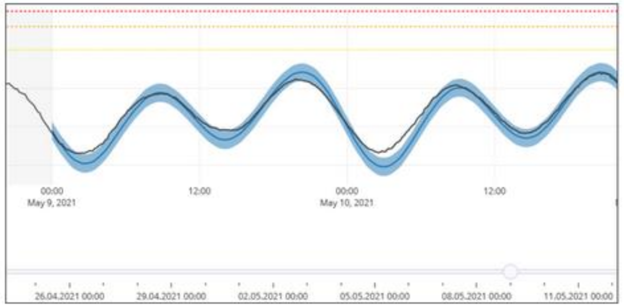
# About the lecturer

- Name: Matej Kristan
- Where to find me: 2nd floor, ViCoS (not in office, in the lab most of time)

- Online contacts and resources:
  - www.vicos.fri.uni-lj.si/matejk
  - ResearchGate
  - Google Scholar
  - eclassroom (https://ucilnica.fri.uni-lj.si/)
  - mail:matej.kristan@fri.uni-lj.si

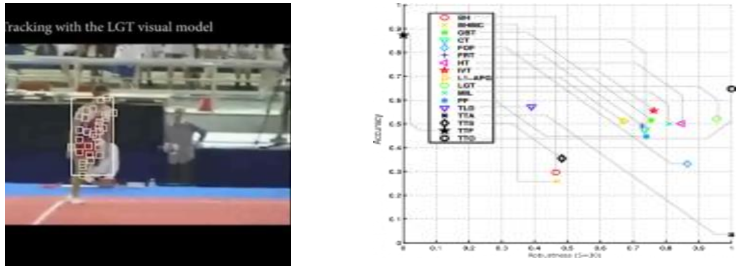
1. Industrial vision, anomaly, counting

2. ML for geophysics (HIDRA link)



3. Robotic vision


4. Visual object tracking



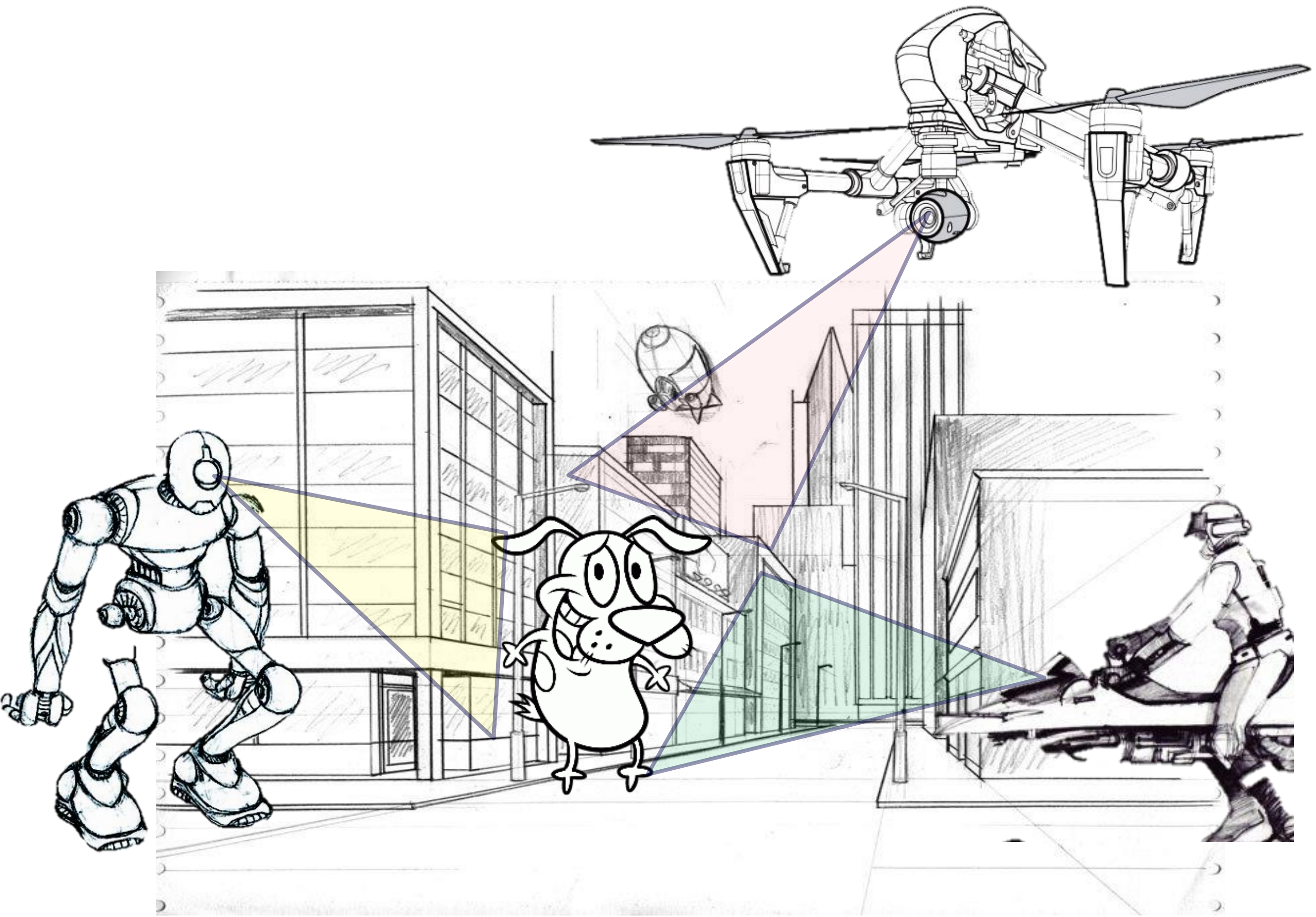
# What will this course be about?

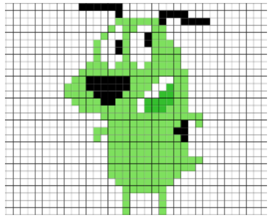
- Motion perception and Tracking

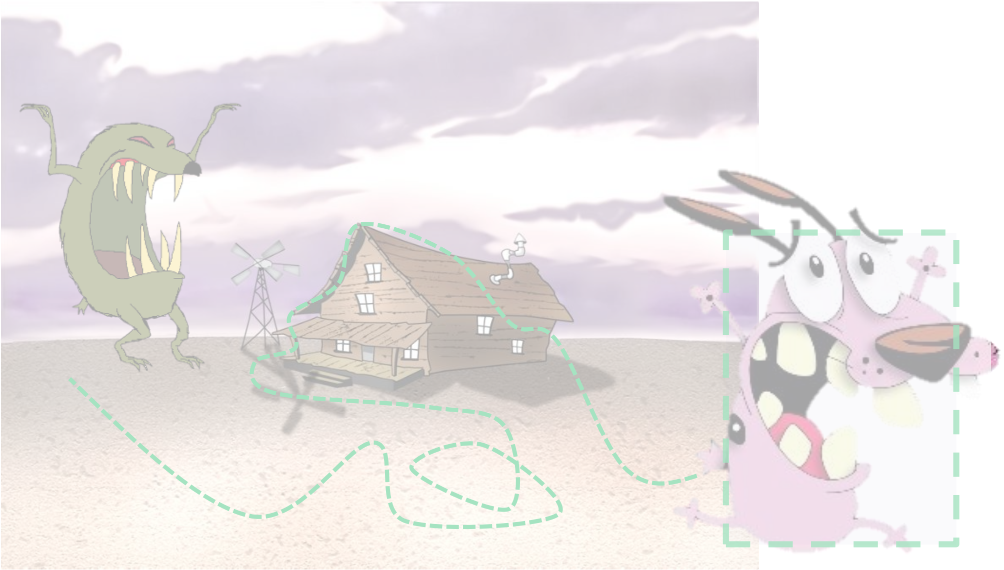


- Currently hot topics in CV as well as industry

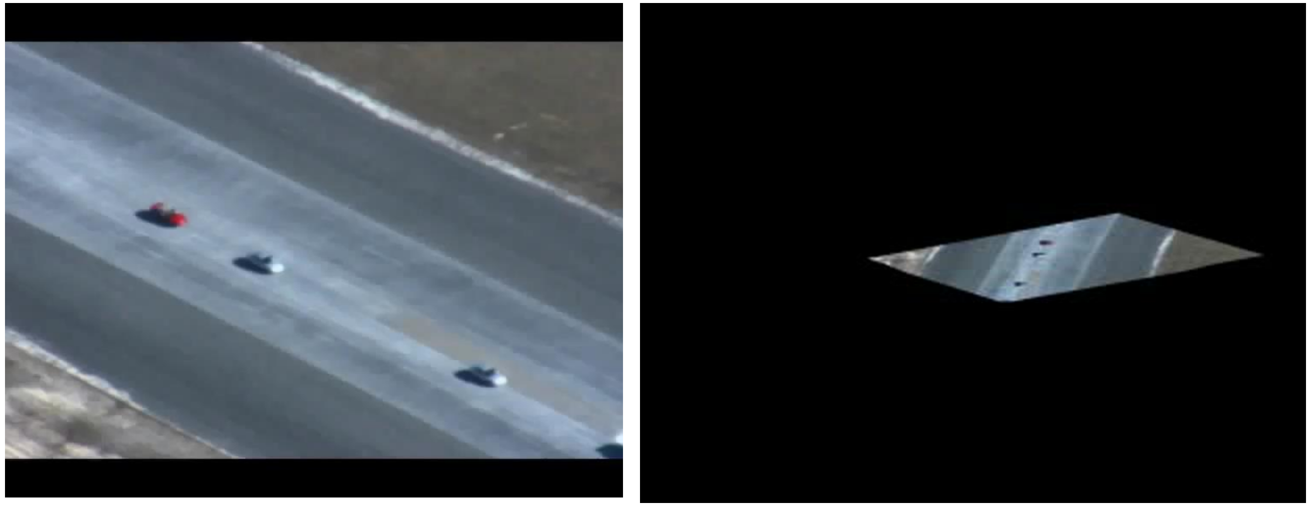
# A huge application potential





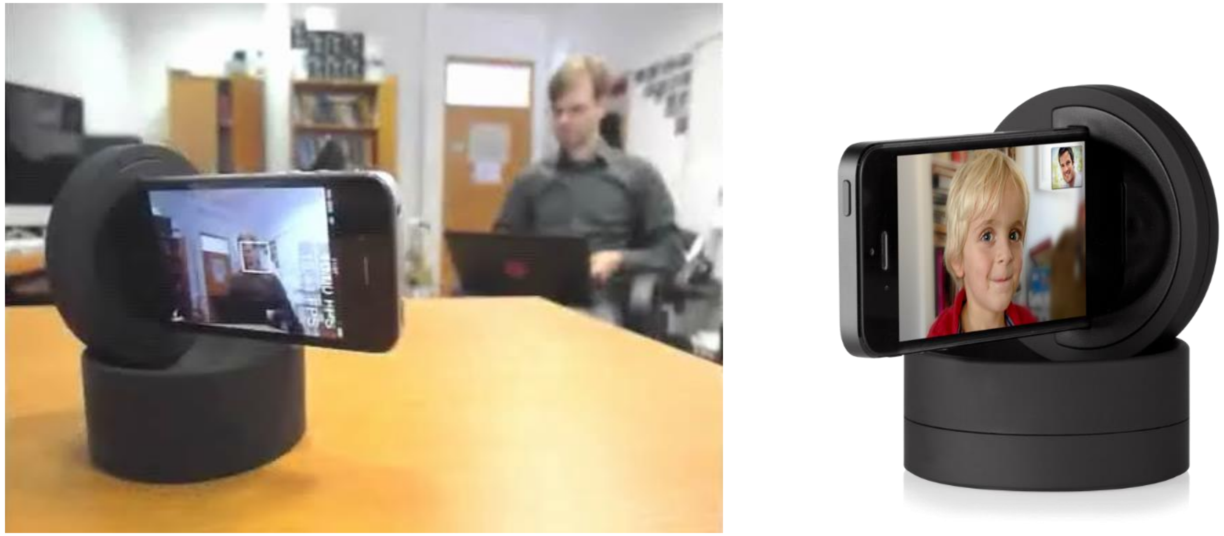


# Application examples



SaadAli , Mubarak Shah ISR2006

Perazzi et al., CVPR 2016



Čehovin, Kristan and Leonardis, IEEE TPAMI 2013


Kristan et al. CVIU 2009

Advanced Methods in Computer Vision

# DETAILS ABOUT THE COURSE

# Main topics

1. Low-level motion estimation techniques
2. Tracking regions by generative models (templates)
3. Tracking regions by discriminative models
4. Bayesian recursive filtering
5. Deep-learning-based trackers
6. Long-term tracking
7. Visual tracking performance evaluation

Tentative!
May change

# Required background

- Programming

  (Python experience preferred)

- Basic algebra and vector/matrix calculus

  (basic, but good foundations)

- Basic probability and statistics

  (basic, but good foundations)

- Basics in signal processing / computer vision desired

  (will provide the references to the relevant literature)

# Preliminaries on deep learning

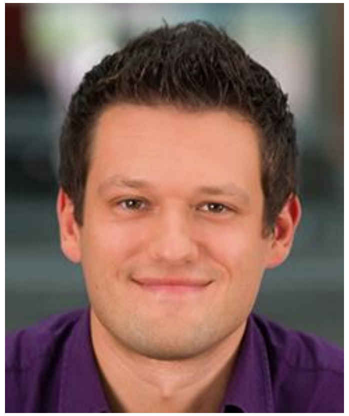
- Deep learning is an elementary methodology in computer vision
- Towards the end of semester a lecture on trackers based on CNN
- You are required to be familiar with general neural networks and have a grasp of the basic ideas behind the CNNs.
- If you're not familiar, familiarize yourself:
  **CS231n: Convolutional Neural Networks for Visual Recognition**
  http://cs231n.stanford.edu/schedule.html
  - Lecture 4 (basics of neural nets)
  - Lecture 5 (convolutional neural networks)
  - Lecture 6 - Lecture 9 (training the networks and some relevant architectures)

# Lectures

- First-hand insights on the topics
  - Ask questions!
- Will cover main concepts and go over the necessary derivations
- Attend the lectures and make your own notes!
- Literature/resources:
  - Lecture slides (void of derivations)
  - Recordings of Covid-year lectures available on MS Stream (but increasingly out of date!)
  - Major conference or journal papers

# Practicum (Projects)

- Starts in 3rd week of semester
- Goal: Learn the theory by implementing it!
- Implementations in Python



dr. Alan Lukežič

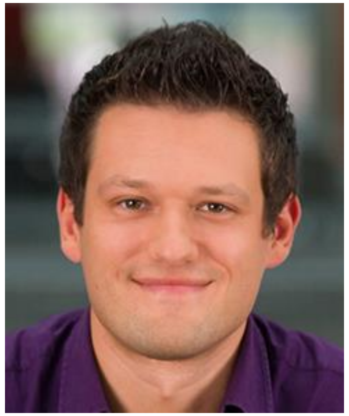
mag. Jer Pelhan

- Complete 5 assignments (start in 3rd week)
  - Implement what you learned at lectures
  - Two-week assignments, brief guidelines, individual work required
  - No formal in-person classes, but consultations available.

    (more on next slide)

# Practicum (Projects)

- Assignment opening and submissions:
  - Assignment instructions appear on e-classroom, along with a short video explaining the task
  - Submisson closes in two weeks on Sunday (23:59)
    - Submit a short report & code
    - Code will sporadically be checked.  Detection of plagiarism will have dire consequences!
  - Important: Start early and consult the assistants!



dr. Alan Lukežič

mag. Jer Pelhan

- Consultations: main contact – Alan Lukežič
  - Ask by email or arrange consultations in person (alan.lukezic@fri.uni-lj.si)
  - Ask on MSTeams channel and/or arrange an online meeting (link)
  - Disregard the “urnik” – ping the assistant anytime

# ACVM course Gantt diagram

13 lectures (Tentative cancellation due to May holiday)

~ 4-6 non-mandatory home works (lightweight)


Assignments (two weeks effort!)

Assignment1 Assignment2 Assignment3 Assignment4 Assignment5

# The A, B and C of the course

- A: 5 lab assignments / practicum
  - Further details at the lab
- B: a few homework assignments
  - To help you follow the lectures
  - Only the first one will be graded
- C: Written exam
  - Mainly theory + basic computations

Final grade: A*0.6 + C*0.4
(first HW will contribute to the points in the assignment 1)

NOTES:
Positively pass all assignments in (A) – required
Pass the written exam (C) – required
Homework (B) – not required, but desired

# Where to find state-of-the-art?

## Three top journals of CV: (Source: Cobiss.si)

- IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE TPAMI
- IEEE TIP ; Pattern Recognition ; IJCV

## Top conferences:

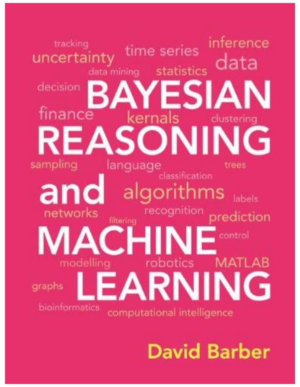
(Source: Microsoft academic research)

| |
|---|
| CVPR - Computer Vision and Pattern Recognition |
| ICCV - International Conference on Computer Vision |
| ECCV - European Conference on Computer Vision |
| FGR - IEEE International Conference on Automatic Face and Gesture Recognition |
| BMVC - British Machine Vision Conference |
| WACV - Workshop on Applications of Computer Vision |
| ACCV - Asian Conference on Computer Vision |

# Some textbooks/handbooks

- General CV: Computer Vision Models Learning & Inference
  - (freely available)
  - Linear algebra: Appendix C


- Probability: Bayesian Reasoning and Machine Learning
  - (freely available)
  - Vectors, matrices, gradients: Appendix A




- Some readily computed matrix-vector derivatives
  - The matrix cookbook (freely available)

# Today – getting on the same page

- Have a look at linearization
  - Most of you should be familiar with this, but I will not assume that

- Get some homework (4 exercises)

  !! POINTS COUNT FOR ASIGNMENT 1!!
  - Turn in the homework by next week (see the e-classroom for exact date)
  - Submit via e-classroom

- Promise more computer vision fun in the following lectures ☺

Advanced computer vision methods

# LINEARIZATION IN A NUTSHELL

# A task often encountered

- Have a parametric model.
- Find parameters of the model to best fit the data.

- A fitting example:

  By how much should be expand/shirk Stan to best fit Olio?

# Parameterized Stan’s face

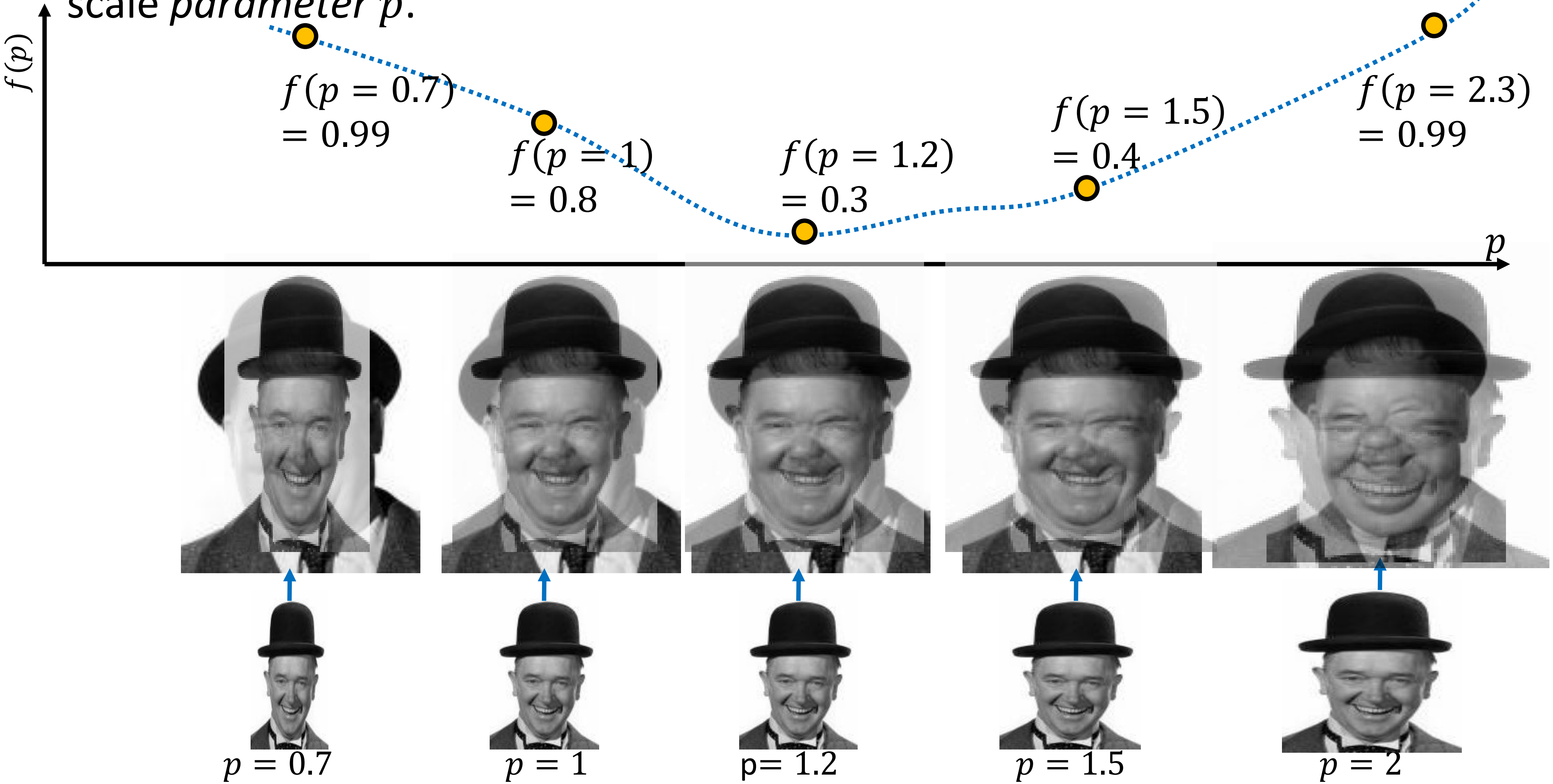
- Stan’s width parameterized by a scale factor $p$, i.e., the new image width is $W_{new} = W_{old} \cdot p$

$p = 0.7$ $p = 1$ $p = 1.2$ $p = 1.5$ $p = 2$

- Now we need to compare Stan’s warped face to Olio’s face…

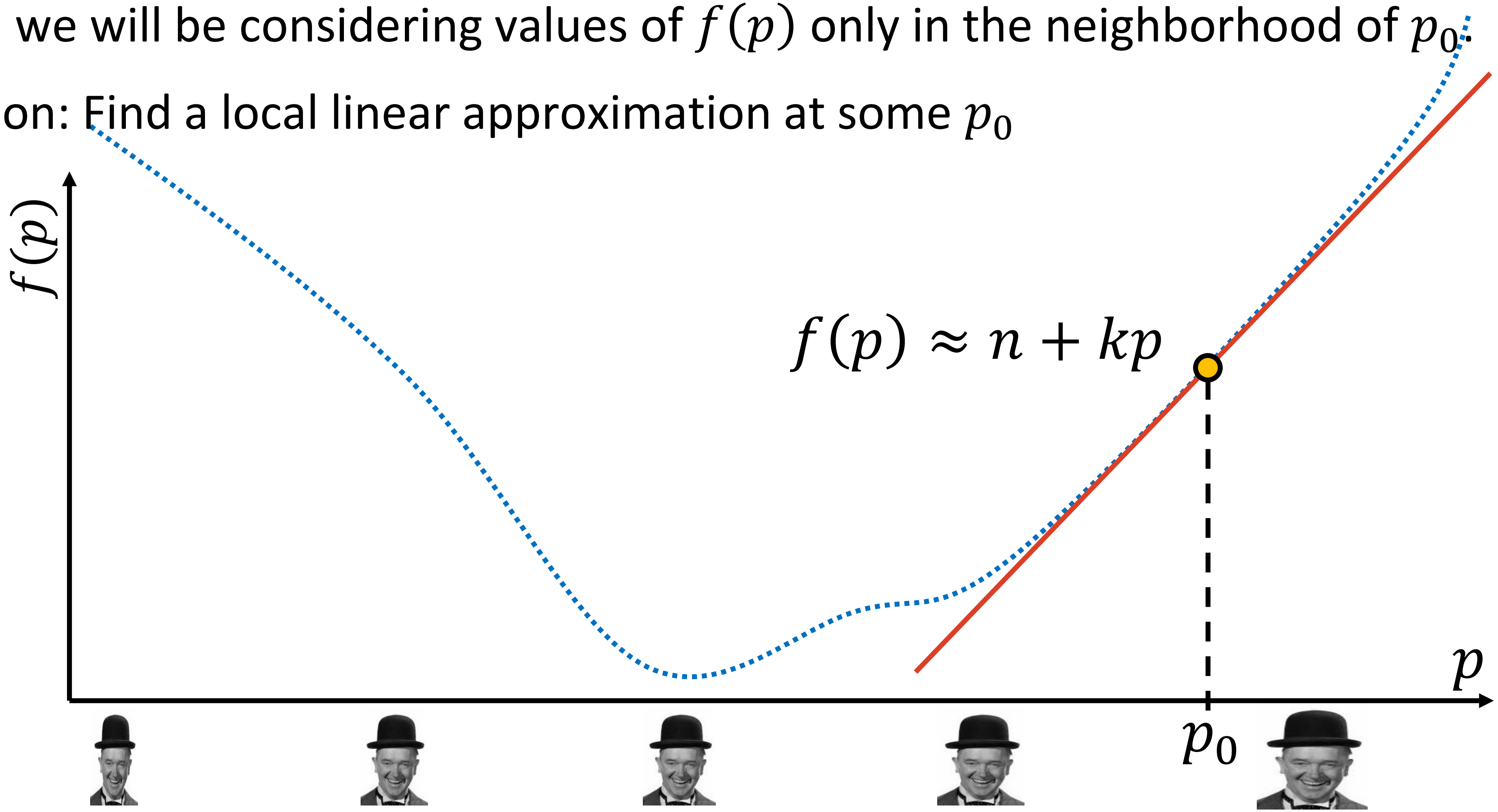
# Parameterized Stan's face

- Difference $f(p)$ between Stan's face deformed by $p$ and Olio's face depends on the scale *parameter* $p$.



# Parameterized Stan's face

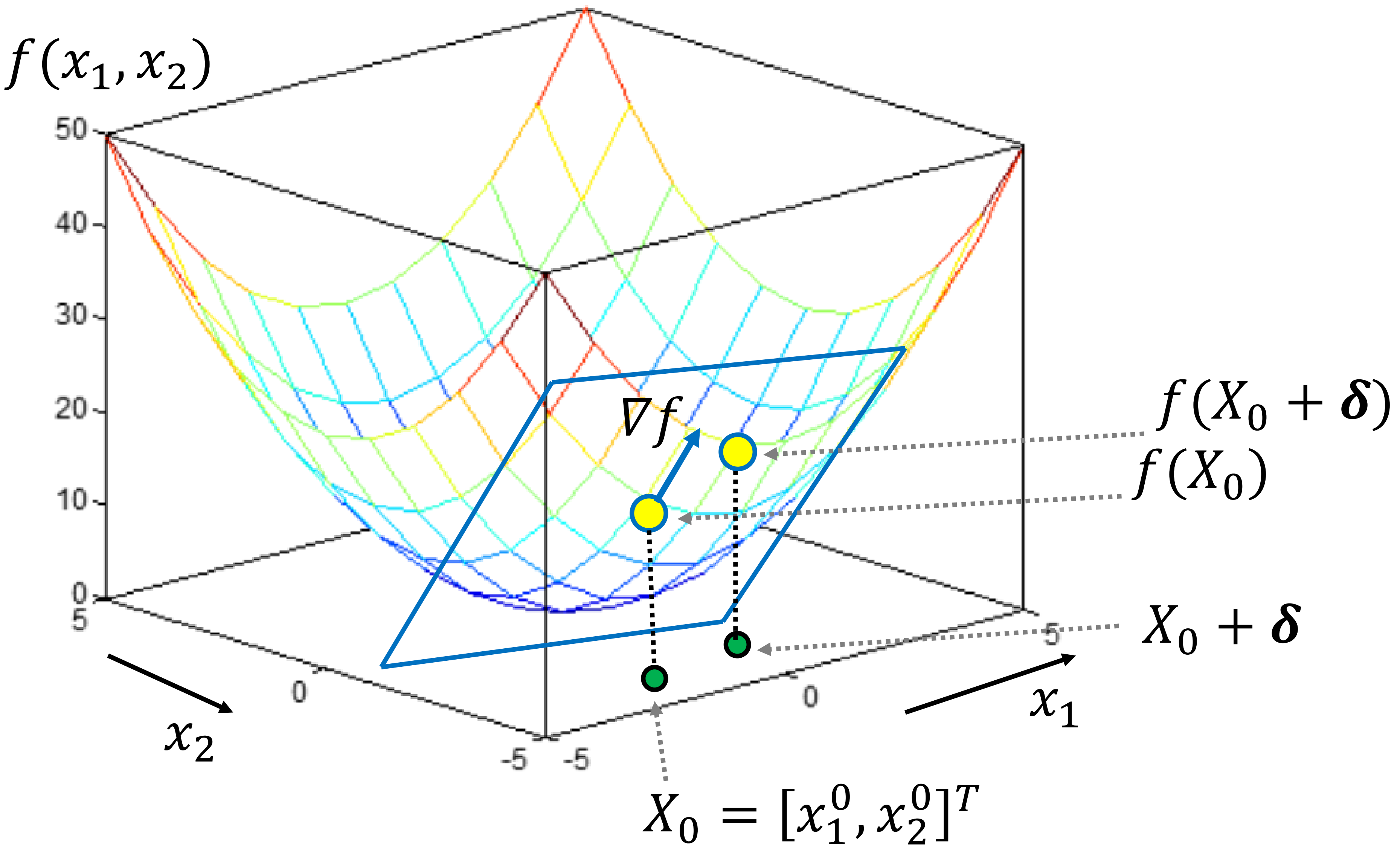
- Often we will want to use the function $f(p)$ in our computations, but working with nonlinear functions can complicate calculations.
- Often we will be considering values of $f(p)$ only in the neighborhood of $p_0$.
- Solution: Find a local linear approximation at some $p_0$



# General problem emerges

- Given a nonlinear function $f(x(\boldsymbol{p}))$ parameterized by some parameters $\boldsymbol{p} = [p_1, p_2, \dots, p_n]$, what is the linear approximation at a neighborhood of parameters $\boldsymbol{p_0}$?
- Linearize by Taylor expansion
  (ignore higher-order terms).
- See notes that you took at lectures.

# Multivariate gradient



# Linearization by Taylor expansion

- To brush up on Taylor expansion and linearization, see "Bayesian Reasoning and Machine Learning" Appendix A, Section 29.2
- For explanation of the gradient and partial derivatives, specifically, equation (29.2.4) for linearization by Taylor expansion.
- Interactive examples of multivariate derivatives: http://mathinsight.org/linear_approximation_multivariable