Erasmus Mundus Joint Master Degree in Image Processing and Computer Vision (EMJMD-IPCV)

Image Processing AND Computer Vision

Applied Video Sequence Analysis
Lab 3 “Kalman filtering for object tracking”

Instructor: Juan Carlos San Miguel (Juancarlos.Sanmiguel@uam.es)

Teaching assistant: Paula Moral (Paula.Moral@uam.es)

PÁZMÁNY PÉTER CATHOLIC UNIVERSITY

UAM Universidad Autónoma de Madrid

université de BORDEAUX

OBJECTIVES

- **Develop two algorithms for object tracking based on the Kalman filter framework** using the OpenCV C++ library

- **Analyze the strengths and weaknesses** of the different modules that compose the pipeline of the algorithms

- **Increase independency and self-learning skills**, guidance decreases as compared to previous assignments.

DELIVERY RULES

- Assignment available on Moodle to submit your material
- The material must be submitted as a ZIP file with the following format *Surname1name1_ Surname2name2_lab3.zip*
- The submitted ZIP file will contain
  - Report in PDF format following the guidelines (max 10 pages)
  - For each task, a directory containing:
    - *Makefile* to compile and link the program by simply running *make* *(Suggestion: use the makefile provided for lab1)*
    - "src" directory with all source files (.h, .hpp, .c and .cpp) necessary for compiling and executing the corresponding program in Linux
    - **Please do not submit configuration files of Eclipse**

TASKS

Three tasks

3.1 Based on Background Subtraction: code implementation

3.2 Based on Background Subtraction: analysis of toy data

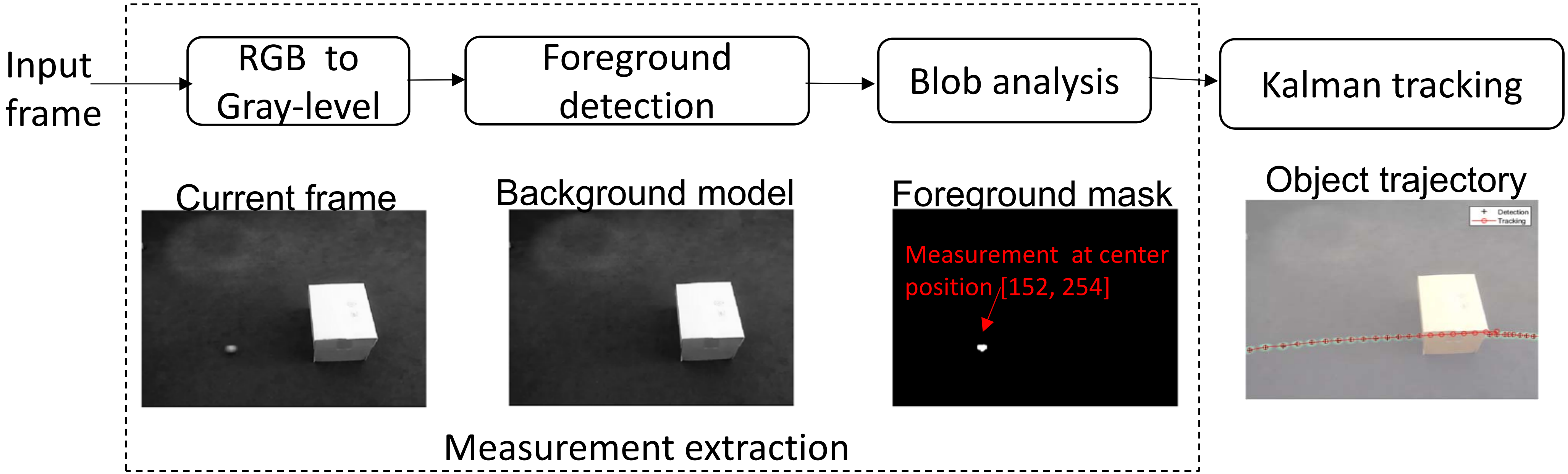
3.3 Based on Background Subtraction: analysis of real data

3.1 Based on Background Subtraction: code implementation

3.2 Based on Background Subtraction: analysis of toy data

3.3 Based on Background Subtraction: analysis of real data

Objective: implement a single-object tracker based on background subtraction and Kalman filtering like the one described in lectures



TASKS

3.1 Based on Background Subtraction: code implementation

3.2 E ta

3.3 E ata

You should get the same results as shown in lectures to validate the correctness of your implementation.

The input video sequence must be an argument of the main function
(similarly as the "Project sample OpenCV 3.4.4 (Linux)" for lab0 available on Moodle)

Mandatory requirements for extracting the measurement:

- Use OpenCV MOG for foreground detection (see https://bit.ly/33BBbAJ)
  (suggested parameters for MOG: learning_rate=0.001, varThreshold=16 and history=50)
- Apply morphological opening to filter noise from the foreground mask
  (suggested parameters for Opening: size=3x3, type=MORPH_RECT)
- Apply blob detection (developed for lab2) to the filtered mask and get as measurement (for Kalman) the center of the biggest blob found in the mask
  (suggested parameters for blob extraction: min width=10, min height=10)

Mandatory requirements for Kalman filtering:

- Use the OpenCV KalmanFilter class (see https://bit.ly/2YOxEN0)
- Implement two models: constant velocity and constant acceleration
- An example of how to use KalmanFilter https://bit.ly/2WHCOs8

Test sequences: "singleball.mp4"

TASKS

3.1 Based on Background Subtraction: code implementation

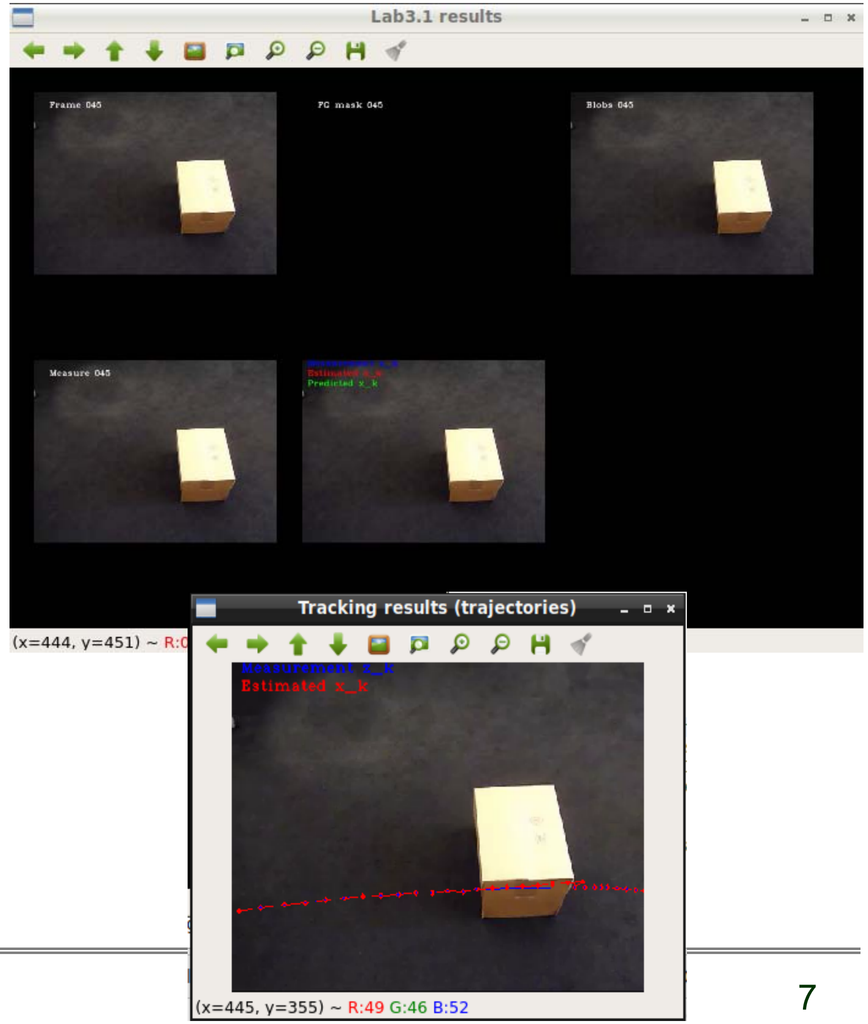
3.2 E … ta

3.3 E … ata

You should get the same results as shown in lectures to validate the correctness of your implementation.

Considerations for the program output:

- For rest of the tasks, it would be convenient to plot the coordinates for all frames of measurements, predictions and estimations of the object state
- Suggestions (optional):
  - Use std::vector class for accumulation https://bit.ly/395HDRY (e.g. lists of cv::Point as std::vector<cv::Point>))
  - Use cv::putText to plot text over images
  - Use the ShowManyImages class to display many images at once
  - Avoid large portions of code in the main function
  - Use/create classes and methods whenever possible
  - Add comments to your code (at least functional units/methods)



TASKS

3.1 Based on Background Subtraction: code implementation

3.2 Based on Background Subtraction: analysis of toy data

3.3 Based on Background Subtraction: analysis of real data

Objective: analyze the effect of Kalman components for toy video sequences while keeping fixed measurement extraction parameters

(parameters for MOG: learning_rate=0.001, varThreshold=16 and history=50; parameters for Opening: size=3x3, type=MORPH_RECT; parameters for blob extraction: min width=50, min height=50)

Starting from the code developed in task 3.1, visually explore differences in tracking performance (i.e. how good is the estimated trajectory)…

1) …between the constant velocity and acceleration models. Why one model performs better than the other?
2) …when measurement noise covariance (R matrix) is changed from the default value seen in lectures. Why changing R gets different results?

Test sequences: “video2.mp4”, “video3.mp4”, “video5.mp4” and “video6.mp4”

3.1 Based on Background Subtraction: code implementation

3.2 Based on Background Subtraction: analysis of toy data

3.3 Based on Background Subtraction: analysis of real data

Objective: analyze and tune the Kalman-based tracker in real video sequences from changedetection.net

Starting from the code developed in task 3.1, …

1) Analyze the tracking problems that exist for all sequences
2) Adjust the parameters of the system to track the single object appearing in each video sequence.

(suggestion: start from the parameters of task 3.2 and change them to get accurate measurements; then focus on adjusting the Kalman filter parameters)

Test sequences: “boats_6950_7900_clip.mp4”, “pedestrians_800_1025_clip.mp4”, “abandonedBox_600_1000_clip.mp4” and “streetCornerAtNight_0_100_clip.mp4”

TASKS

- Expected workload (~20 hours total)

TASK	Expected hours	Type of work*
Task 3.1	8 (40%)	Read and understand material/tasks of lab3 Source code
Task 3.1	2 (10%)	Description of the code in the report
Task 3.2	5 (25%)	Description/analysis in the report
Task 3.3	5 (25%)	Description/analysis in the report
TOTAL	20 (100%)	

EVALUATION

- This lab assignment will be **graded with 10 points**

Concept evaluated	TASK	Max. Score	Criteria evaluated[1]
Source Code	**Task 3.1**	**4**	**Code: Functional requirements (60%)** **Code: Design & structure (30%)** **Code: Style (10%)**
Report	**Task 3.1**	**1**	**Report: structure (20%)** **Report: introduction & methods (80%)**
	Task 3.2[2]	**2.5**	**Report: Experimental methodology[3] (15%)** **Report: Analysis & discussion[4] (25%)**
	Task 3.3[2]	**2.5**	**Report: Experimental methodology[3] (15%)** **Report: Analysis & discussion[4] (25%)**
	TOTAL	**10**	

[1]Criteria for evaluation described in the rubric available in Moodle

[2]For each task, it is suggested to create independent sections in the report for Experimental methodology and Analysis & discussion of tasks 3.2 & 3.3.

[3]In this part please describe the video sequences you are using, their challenges and how you are going to proceed for the analysis and discussion (methodology)

[4]In this part please comment on the results that you see for the tested video sequences and the reasoning behind actions you have taken to improve performance.

EVALUATION

- This lab assignment will be **graded with 10 points**

Concept evaluated	TASK	Max. Score	Criteria evaluated[1]
Source Code	**Task 3.1**	**4**	**Code: Functional requirements (60%)** **Code: Design & structure (30%)** **Code: Style (10%)**
Report	**Task 3.1**	**1**	**Report: structure (20%)** **Report: introduction & methods (80%)**
	Task 3.2	**2.5**	**Report: Experimental methodology (15%)** **Report: Analysis & discussion (25%)**
	Task 3.3	**2.5**	**Report: Experimental methodology (15%)** **Report: Analysis & discussion (25%)**
	TOTAL	**10**	

- Penalties:
  - Delivery not following requirements: -0.5 points
    (examples are: submission without makefile; report not following the course format;)
  - Late delivery after the remaining days of the late policy (remember 4-days in total for all labs)
    - -25% (one day), -50% (two days), -75% (three days), -100%(>= four days)

HELP

- Check OpenCV documentation for finding specific functions at https://docs.opencv.org/3.4.4/