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

Image Processing and Computer Vision

Applied Video Sequence Analysis

Lab 1 “Foreground segmentation”

José M. Martínez - Paula Moral

josem.martinez@uam.es - paula.moral@uam.es

Universidad Autónoma de Madrid

Escuela Politécnica Superior

Video Processing and Understanding Lab

OBJECTIVES

- **To develop a basic foreground segmentation algorithm** to detect moving objects on a stationary background
- **To improve the performance** of the basic algorithm
- **To evaluate the performance of foreground segmentation** algorithms using standard measures and Matlab code

- **To write a report** to summarize achievements and results
  - Motivate the development of the different versions and modes (i.e., what does it enhance?)

DELIVERY RULES

- Assignment available on Moodle to submit your material

- The material must be submitted as a ZIP file with the following format *name1Surname1_ name2Surname2_lab1.zip*

- The submitted ZIP file will contain
  - Report in PDF format (max 6 pages)
  - *Makefile* to compile and link the program by simply running *make** *(Suggestion: use the makefile provided for Lab0)*
  - “src” directory with all source files (.h, .hpp, .c and .cpp) necessary for compiling and executing the corresponding program in Linux
  - “res” directory with the foreground masks for the *Baseline* category of CDNet2012 for different algorithms versions implemented.

*You may want to develop your C/C++ program using Eclipse but only the Makefile must be submitted (please do not submit Eclipse’ config files)

TASKS

- Task 0 – Void foreground segmentation using template
  1. Go to “LAB 1” on Moodle
  2. Download the example project “LAB 1 code template.zip”
  3. Download the datasets “AVSASlidesVideosdataset” and “ChangeDetection dataset 2012 (baseline category only)”
  4. Uncompress the files
  5. Create a new Project Lab1.0AVSA2020 in Eclipse (see next slide)
  6. Open the Lab1.0AVSA2020.cpp file in the “src” directory
  7. Set the directory of the dataset and results according to your installation
  8. Compile it (“Build Project” in Eclipse)
  9. Create a “Launch Configuration” for the Project
     - Select the Project in the “Project Explorer” tab, go to the toolbar and select in the “Launch Configuration”pane “Create New Launch Configuration”
     - In the “Environment” tab set ***LD_LIBRARY_PATH=/opt/installation/OpenCV-3.4.4/lib***
  10. Run it …

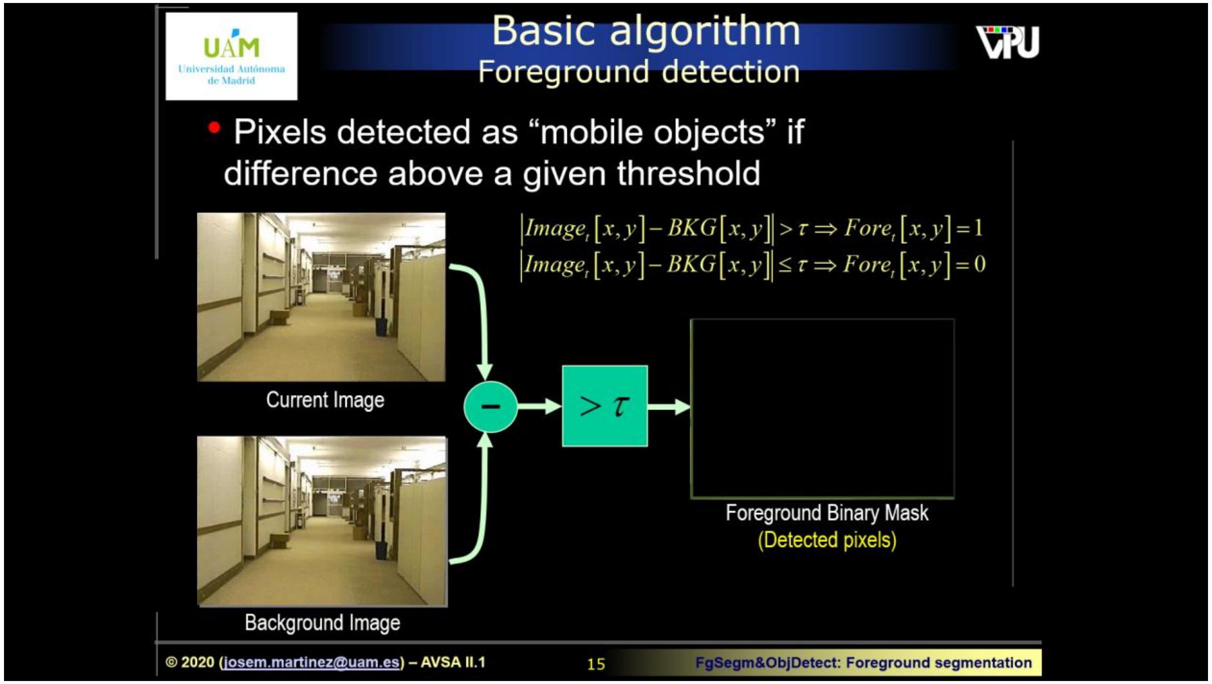
TASKS

- Creating an Eclipse Project from scratch (see Lab0)
  - How to install and setup Eclipse for OpenCV 3.4.4
    - Configure Eclipse CDT for using OpenCV
- Creating an Eclipse Project with existing code
  - After creating the Projects and src folder …
  - … instead of creating a new source file (step 7), copy the files (.cpp, .hpp) into the src folder
    - Select them in a Files explorer and move them to the Project src folder in Eclipe
  - Continue with step 8
- …

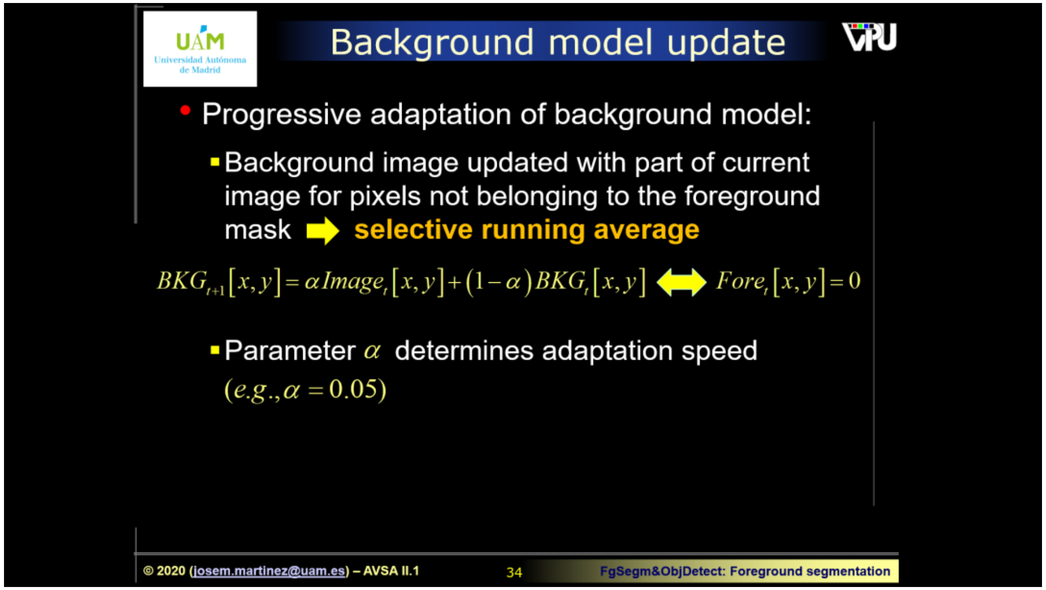
TASKS

- …
- Creating a copy of an Eclipse Project (for incremental development – e.g., different versions in Lab1)
  - Select source Project in “Project Explorer”: then “copy” and “paste”
  - Name the new project
  - Rename “main” file and change *project dependent* code (e.g., project_name)
  - Delete Binaries in Project Folders (e.g., Debug)
  - Build the Project
  - Create Launch Configuration for the new project (see Lab0)
    - Remember to set LD_LIBRARY_PATH
  - Run the Project

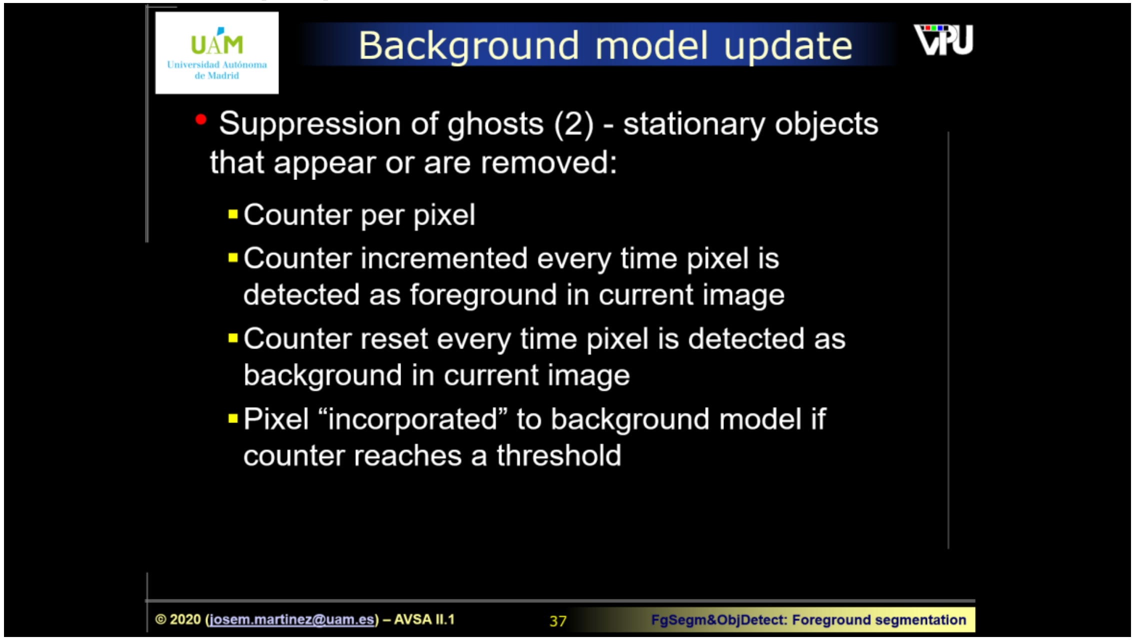
- Task 1 - Implementation of foreground segmentation algorithms using the OpenCV library
  - Methods
    - **Lab1.1.1: Generation of foreground segmentation mask for each video frame using "frame difference"**
      - Parameters: tau (double), rgb (true/false)



- Task 1 - Implementation of foreground segmentation algorithms using the OpenCV library
  - Methods
    - **Lab1.1.2: Progressive update of background model through selective running average (blind and selective update)**
      - Parameters: tau (double), rgb (true/false), alpha (double), selective_update (true/false)
      - *Recommendation: firstly implement blind mode*



- Task 1 - Implementation of foreground segmentation algorithms using the OpenCV library
  - Methods
    - **Lab1.1.3: Suppression of stationary objects that appear or are removed**
      - Parameters: tau (double), rgb (true/false), alpha (double), threshold (int)

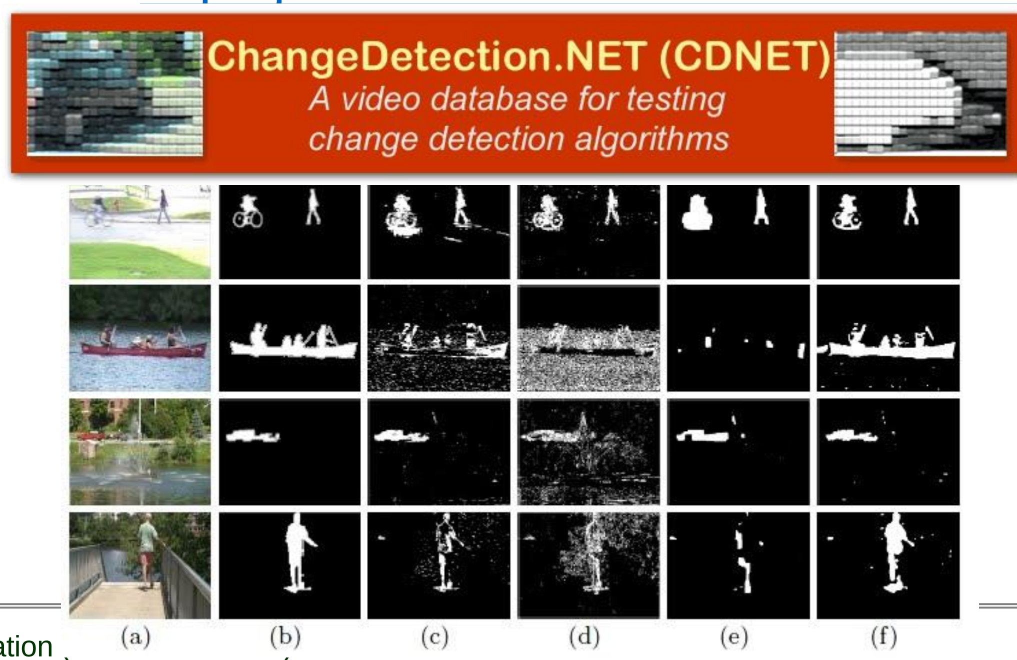


TASKS

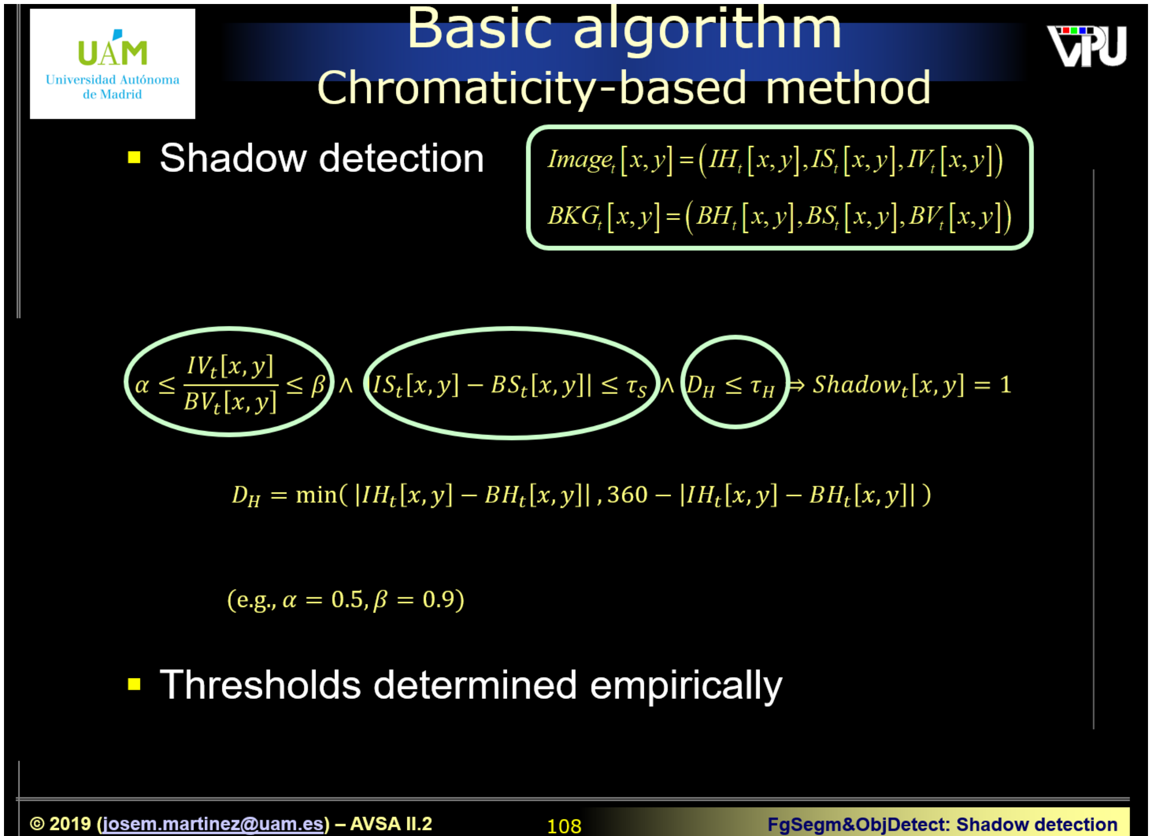
- Task 1 - Implementation of foreground segmentation algorithms using the OpenCV library
  - Requirements
    - Only use three source files (main.cpp, fgseg.cpp and fgseg.hpp)
    - The algorithm operates over gray-level images and colour-images (optional)
    - Report comparative subjective results using the VideoSlidesAVSA dataset for the three versions
    - Report a sensitivity analysis of the parameters involved in algorithms (i.e. study the values of the parameters involved and their impact in the output or the foreground mask)
  - Recommendations
    - Develop firstly the gray-level versions
    - Motivate in the report the development of the different versions and modes (i.e., what does it enhance?)
    - Provide a subjective comparative among versions using the AVSASlidesVideo dataset

TASKS

- Task 2 – Evaluate your algorithm using the ChangeDetection datasets (2012 version, CDNet2012)
  - Evaluate the algorithm developed in Task 1 using the Baseline category from http://changedetection.net/
    - Category of the dataset available in Moodle
    - MATLAB code is available for evaluating the results of your algorithms
  - In the report, include performance comparisons with selected approaches from http://jacarini.dinf.usherbrooke.ca/results2012

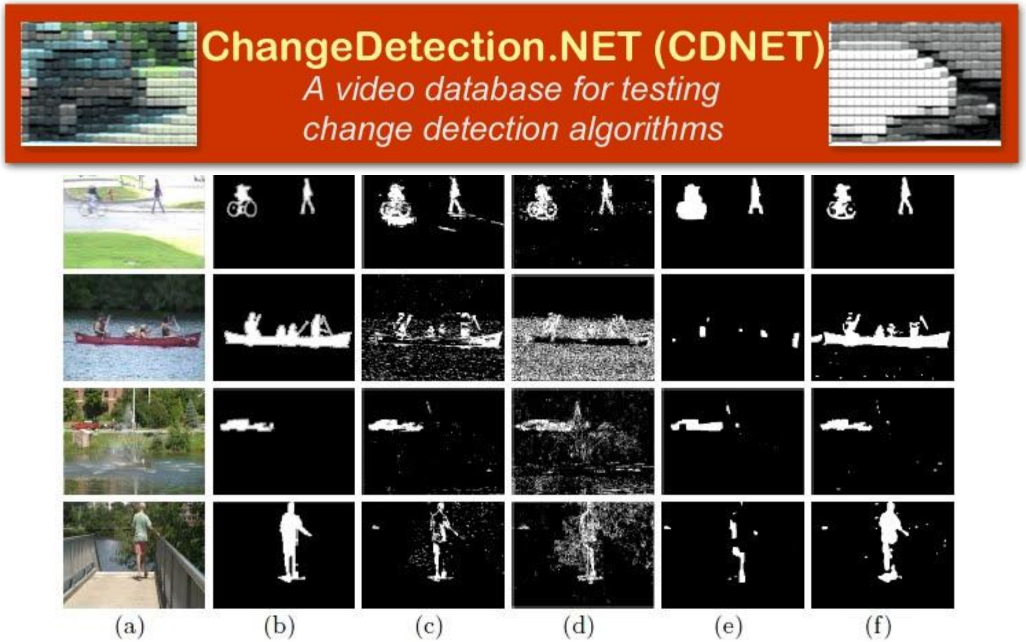


- Task 3.1 - Implementation of a shadow suppression module using the OpenCV library
  - Method
    - Suppression of shadows based on the chromaticy-based method

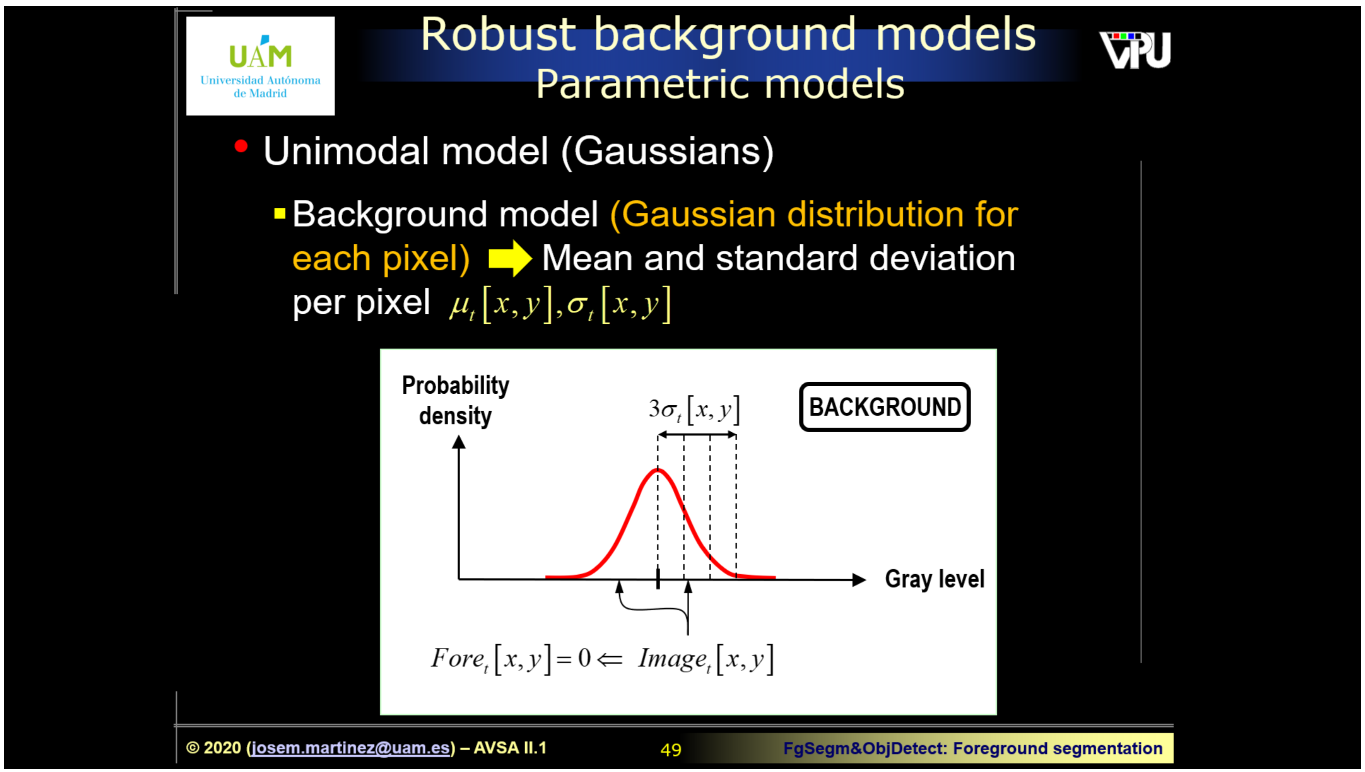


TASKS

- Task 3.2 – Evaluate the suppression of shadows method
  - Evaluation of new algorithm using the *Shadow* category from http://changedetection.net/

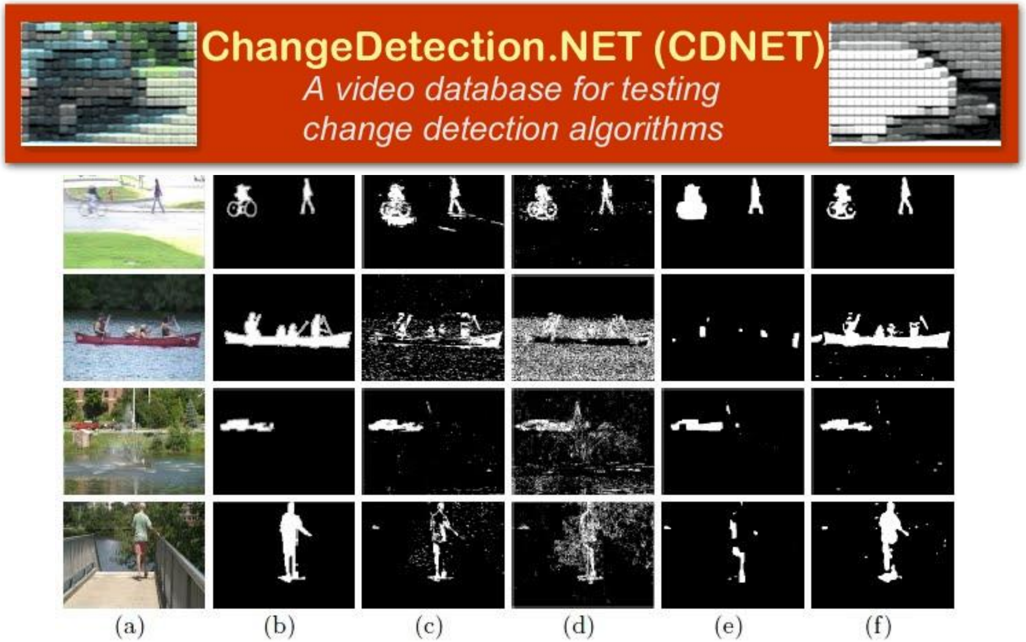


- Task 4.1 – Implementation of advanced background subtraction algorithms
  - Method
    - Single Gaussian

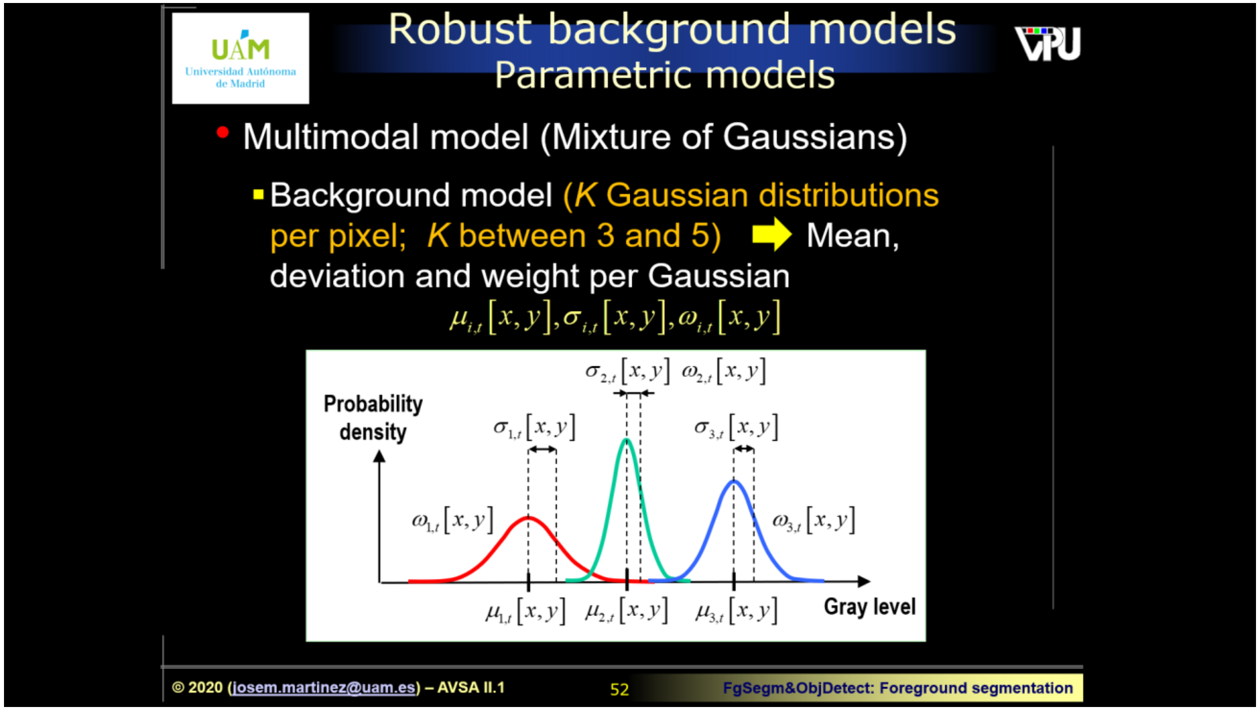


TASKS

- Task 4.2 – Evaluation of advanced background subtraction algorithm
  - Evaluation of the new algorithm using the *Dynamic Background* category from http://changedetection.net/

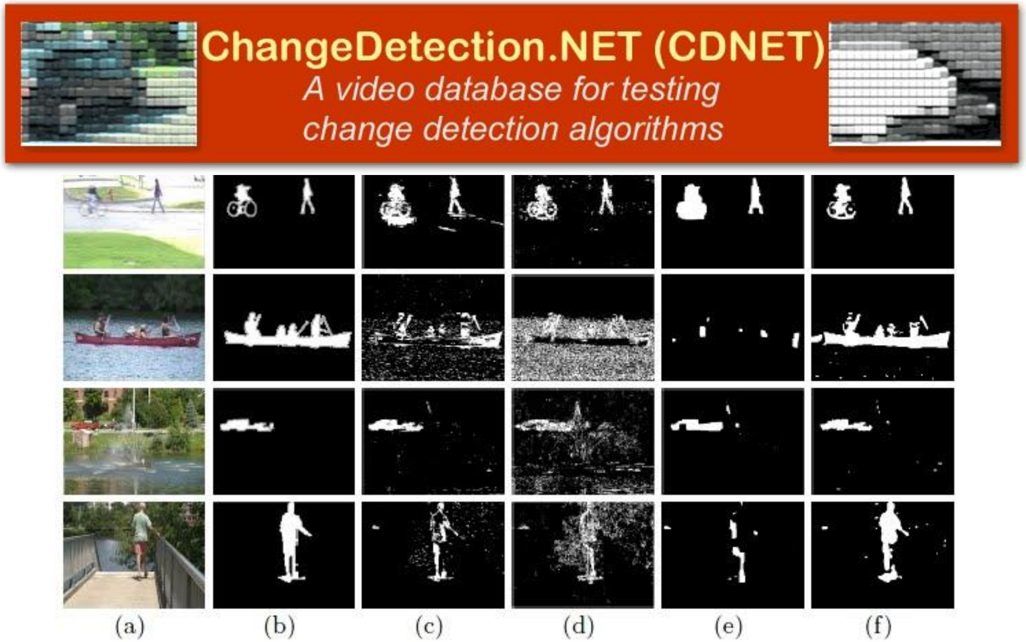


- Task 5.1 – Implementation of advanced background subtraction algorithms
  - Method
    - GMM
  - Recommendation
    - Implement firstly a version using only a Single Gaussian



TASKS

- Task 5.2 – Evaluation of advanced background subtraction algorithm
  - Evaluation of the new algorithm using the *Dynamic Background* category from http://changedetection.net/



EVALUATION

- This lab assignment will be **graded with 10 points**
  - Report (up to 2 points) – Please follow the report format (see *Moodle*)
  - Task 1 (up to 2 points)
  - Task 2 (up to 2 points)
  - Task 3 (up to 1 points)
  - Task 4 (up to 1 points)
  - Task 5 (up to 2 points)

CRITERIA FOR EVALUATION

- The following general criteria will be used for grading:
  - The solution addresses the requirements for the assignment.
  - The program compiles, links, and executes.
  - The program runs correctly.
  - The program is easy to read and to understand, i.e., it is well commented, and variables are correctly named.
  - The lab report clearly describes the algorithm and experiments.

- The following specific criteria will be used for grading:
  - Foreground segmentation using colour information.
  - Complexity of the implemented algorithms
  - Processing with OpenCV functions at matrix/vector level instead of pixel-level.

In addition, up to an extra point (1 point) may be awarded depending on the quality of the software delivered in the assignment.

HELP

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