Goals

- Get a valuable resource for the research community
- Foster and accelerate progress in research and commercial systems

Algorithm competition is unique in that all participating algorithms
- are submitted in source code complying to a minimal, but also very
general C/C++ API based on the Open Source Computer Vision Library
(OpenCV, http://sourceforge.net/projects/opencvlibrary),
- are applied and evaluated on a public data sets, and
- use a performance evaluation metric available in C/C++ source code.

➔ Source code allows open discussion, improvements and reuse.
➔ Enables re-run the tests.
➔ Opportunity to include source code into updates & future releases of OpenCV.
This Year’s Task

**Segmentation of foreground objects**
- Core aspect in many computer vision & surveillance systems
- Provide at every time instance (after maybe some initial training)
  - An estimate of the background image
  - A probability foreground mask
- **Foreground mask** := specifies for each pixel its probability of belonging to a foreground object.
  \[ 8uC1 \text{ image} \]
- **Background image** := how does the current background look like without the foreground object
  \[ 8uC3 \text{ image} \]

Basic Code Loop (Example)

```c
IplImage* tmp_frame = NULL;
CvCapture* cap = NULL;

// capture video from file
cap = cvCaptureFromFile("video.avi");
tmp_frame = cvQueryFrame(cap);

// create BG model
CvBGStatModel* bg_model =
  myCreateFGDSStatModel( tmp_frame );

// for all frames in the video
for( int fr = 1; tmp_frame;
  tmp_frame = cvQueryFrame(cap),
  fr++ )
{
  // update BG model
  bg_model->update( tmp_frame,
  bg_model );

  // show current estimation
  cvShowImage("Background",
  bg_model->background);
  cvShowImage("Foreground Mask",
  bg_model->foreground);

  int k = cvWaitKey(5);
}

// release BG model
bg_model->release( &bg_model );

// release capture
cvReleaseCapture(&cap);
```

© Prof. Dr. Rainer Lienhart, Head of Multimedia Computing, Institut für Informatik, Universität Augsburg
Eichleitnerstr. 30, D-86135 Augsburg, Germany; email: Rainer.Lienhart@informatik.uni-augsburg.de
Common Data Structure

```c
#define CV_BG_STAT_MODEL_FIELDS() "int type; /* type of BG model*/
#define CvReleaseBGStatModel update; /* release function*/
#define CvUpdateBGStatModel update; /* update bg model*/
#define IplImage* background; /* 8UC3 reference background image*/
#define IplImage* foreground; /* 8UC1 foreground image*/
#define IplImage** layers; /* 8UC3 reference background image, can be null */
#define int layer_count; /* can be zero */
#define CvMemStorage* storage; /* storage for "foreground_regions"*/
#define CvSeq* foreground_regions /* foreground object contours*/

/* ignore the variables "int type", "CvMemStorage storage" and "CvSeq*
foreground_regions" */

// define your own model, i.e., extend the CV_BG_STAT_MODEL_FIELDS()
model
typedef struct MyBGStatModel
{
    CV_BG_STAT_MODEL_FIELDS();
    // ... more fields could be added here ...
} MyBGStatModel;
```

Participant

• S. Calderara, R. Melli, A. Prati, R. Cucchiara: “Reliable Background Suppression for Complex Scenes”
• D. A. Migliore, M. Matteucci, M. Naccari: “A Reevaluation of Frame Difference in Fast and Robust Motion Detection”

Last Year:

• F. Porikli, O. Tuzel: “Bayesian Background Modeling for Foreground Detection”
• A. H. Kamkar-Parsi, R. Laganiere, M. Bouchard: “A Multi-Criteria Model for Robust Foreground Extraction”

OpenCV:

• L. Li, W. Huang, I.Y.H. Gu, Q. Tian: “Foreground object detection from videos containing complex background”, ACM Multimedia, 2003”
Test Videos (1)

3 Categories:

- Vacillating background & gradual illumination changes (video 1 & 2)
- Bootstrapping, i.e. a training period of absent foreground is not available (video 3 & 4)
- Sudden illumination changes (video 4 & 5)

Test Videos (2)
Performance Metrics

Precision = \frac{\text{# of correctly detected foreground pixels}}{\text{# of detected foreground pixels}}

Recall = \frac{\text{# of correctly detected foreground pixels}}{\text{# of actual foreground pixels}}

Fps = \frac{\text{# of frames}}{\text{runtime}}

Average Performance

![Graph showing average performance over the last years with the winner being Grimson.](image-url)
Next Year

• Plans