104th European Study Group with Industry
September 23 - 27, 2014, Sofia, Bulgaria

Problem 1: Effective recognition of the video pattern in a recorded video stream

ESGI 104

CROSS Online Bulgarian Network

Dimo Dimov, IICT-BAS
Atanas Nikolov, IICT-BAS
Krassimira Ivanova, IMI-BAS
Maroussia Bojkova, IMI-BAS, FMI-SU
Plamen Mateev, IMI-BAS, FMI-SU
Ognian Kounchev, IMI-BAS
Vassil Kolev, IICT-BAS
Miroslav Ivanov, IMI-BAS

Outline

1. **Problem formulation and the company's needs**
   1.1. The simplest and the slowest intuitive solution
   1.2. Speed-up, knowing JPEG's DCT coefficients in advance
   1.3. The fastest and entirely problem solving solution
2. Description of the proposed algorithm for full problem solving
   2.1. Forming differential videos (TV stream and advertisements)
   2.2. Scenes detection and videos representation as integer chains
   2.3. Detection of full and/or partial scenes from Adverts in TV
   2.4. Pseudo code of the proposed algorithm
3. Conducted experiments
4. Conclusion words

CROSS Online Bulgarian Network

1. Problem formulation and the company's needs

Input:

Offline (30min) TV video files (*.mp4)

Advertisement video templates (30sec)

Output:

Existing or not of a given Ad template in 24h TV records.

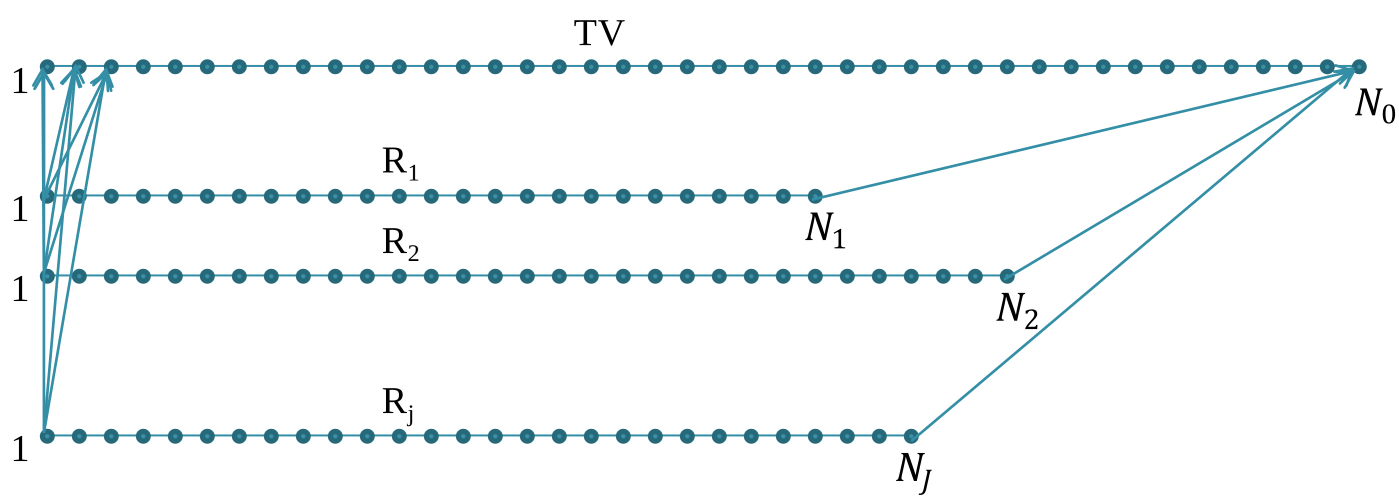
[its approximate time location in the TV stream]

CROSS Online Bulgarian Network

1.1. The simplest and the slowest intuitive solution

- The simplest, but the slowest method for checking if a frame/image (or a part of it) of a given template video is contained in another video stream can be done by direct pixel comparison frame-by-frame from the two video sets.

- Thus, the maximum number of comparison will be $N_0(\sum_{j=1}^{J} N_j)$ where N_0 is the number of frames in the video stream, and N_j are number of frames in the *j-th* template video.



1.2. Speed-up, knowing JPEG's DCT coefficients in advance

- Using DCT (Discrete Cosine Transform) coefficients of 8x8 region(s) from the JPEG frames, some acceleration in the btw-frame comparison function could be achieved.
- This is true, because many of the DCT coefficients in 8x8 regions are zeros, and we will compare less numbers than pixels in the same region.
- In spite of this, the maximum number of comparison remains as in 1.1.
- Furthermore, using DCT coefficients would have sense if we could extract them directly from JPEG frames, but not computing them again.
- In Matlab, there is such a function for accessing DCT coefficients from a JPEG file, but we haven't tried this variant yet (maybe because of the big number of comparisons ?!).

1.3. The fastest and entirely problem solving solution

- Instead of nearly full 'visiting' all the frames from the templates and video stream, we represent the videos as 1D chain of numbers, which characterize the duration (in number of frames) of separate scenes of the videos. We think a scene as a place in a video where there is sharp change between two consecutive frames. Thus, we compare the scenes duration belonging to the videos, and only for the suspicious scenes (which are able to match) we perform one-to-many frames comparison. This strategy has the big benefit of much smaller number btw-frame comparisons, than cases 1.1. and 1.2.
  - An arising problem here is determining the optimal number of scenes detected – in sense than not to have too many false or too many missed scenes. It is not necessary exactly to have the actual borders of all scenes, because our algorithm is noise tolerant in this context and it must work despite this.

1.3. The fastest and entirely problem solving solution

- An (exact?) approach for finding the (right?) borders of the video scenes could be based on the Translational invariant property of the Fourier Transform. This means than the module of FT spectrum will be the same for two translated image. So, based on an assumption that the frames from one and the same scene have common static background and/or it is changed very smoothly by translation, the difference between the modulus of FT coefficients of these frames will be approximately zero, i.e. the frames belong to one scene. We haven't tried this approach for scene detection yet, although it looks very promising for the wished result's quality.
- Because we persuade more than real time performance, we have used a very simple and fast approach for scene detection, which can be improved in the future in terms of automatically choosing the threshold, which determines where the scenes are. It would be great if we manage to do this in a fast and effective enough way.

Outline

1. Problem formulation and the company's needs
    1.1. The simplest and the slowest intuitive solution
    1.2. Speed-up knowing JPEG's DCT coefficients in advance
    1.3. The fastest and entirely problem solving solution
2. **Description of the proposed algorithm for full problem solving**
    2.1. Forming differential videos (TV stream and advertisements)
    2.2. Scenes detection and videos representation as integer chains
    2.3. Detection of full and/or partial scenes from Adverts in TV
    2.4. Pseudo code of the proposed algorithm
3. Conducted experiments
4. Conclusion words

CROSS Online Bulgarian Network

2. Description of the proposed algorithm

2.1. Forming differential videos for the TV stream and the Ads

- Finding of differential TV video clip

$$\mathrm{I}_{\mathrm{diff}}(n) = \frac{1}{X \times Y} \sum\nolimits_{\forall (x,y)} \left| \mathrm{I}_{\mathrm{n}+1}(x,y) - \mathrm{I}_{\mathrm{n}}(x,y) \right|$$

where $n = (0, \dots, N-1)$ are TV video frames;
$\mathrm{I}_{\mathrm{n}}(x,y) = [0, 255]$ is the intensity of a pixel (x,y) from the n-th frame;
$X \times Y$ is the frame size in pixels

- Finding of differential Advertisement video clips

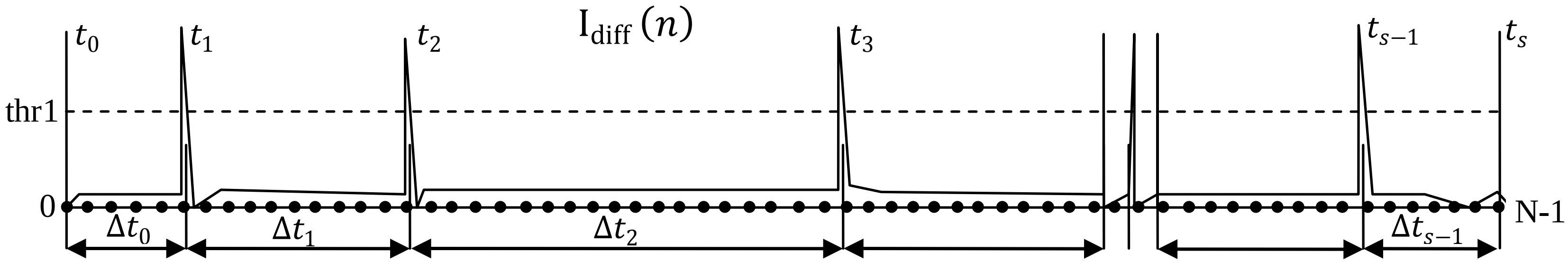
$$R_{\mathrm{diff}}^{j}(l) = \frac{1}{X \times Y} \sum\nolimits_{\forall (x,y)} \left| \mathrm{R}_{l+1}^{j}(x,y) - \mathrm{R}_{l}^{j}(x,y) \right|$$

where $l = (0, \dots, N_j - 1)$ are the frames from j-th Ad
$j = (1, \dots, J) \in \mathcal{R}$ - the set of Ads
R_{l}^{j} is the intensity of l-th frame from j-th Ad

2. Description of the proposed algorithm

2.2. Scenes detection and videos representation as integer chains

- Finding of maximums in the differential TV video clip



if $\mathrm{I_{diff}}(n) > thr1 \rightarrow$ new scene detection $t_i \leftrightarrow n$

$\Delta t_i = t_{i+1} - t_i$ is the length (in frame numbers) of i-th detected scene of the TV video;

$i = (0, \dots, s-1)$, where s is the number of scenes.

The TV chain is defined as: $\mathrm{I_{chain}} = (\Delta t_0, \Delta t_1, \dots, \Delta t_{s-1})$.

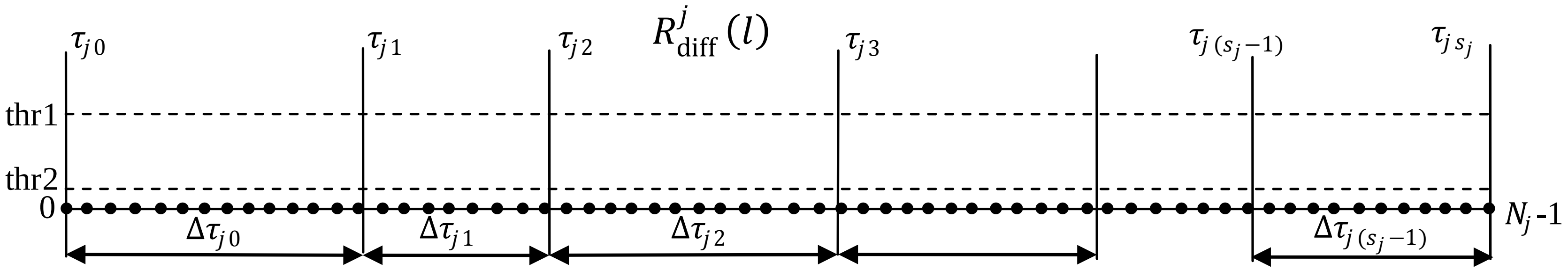
Some additional notes:

- t_i : start of the i-th scene
- $t_{i+1} - 1 = t_i + \Delta t_i - 1$: end of the scene

2. Description of the proposed algorithm

2.2. Scenes detection and videos representation as integer chains

- Finding of maximums in the differential Ad video clips



if $R^{j}_{\text{diff}}(l) > thr1 \rightarrow$ new scene detection $\tau_{ji} \leftrightarrow l$

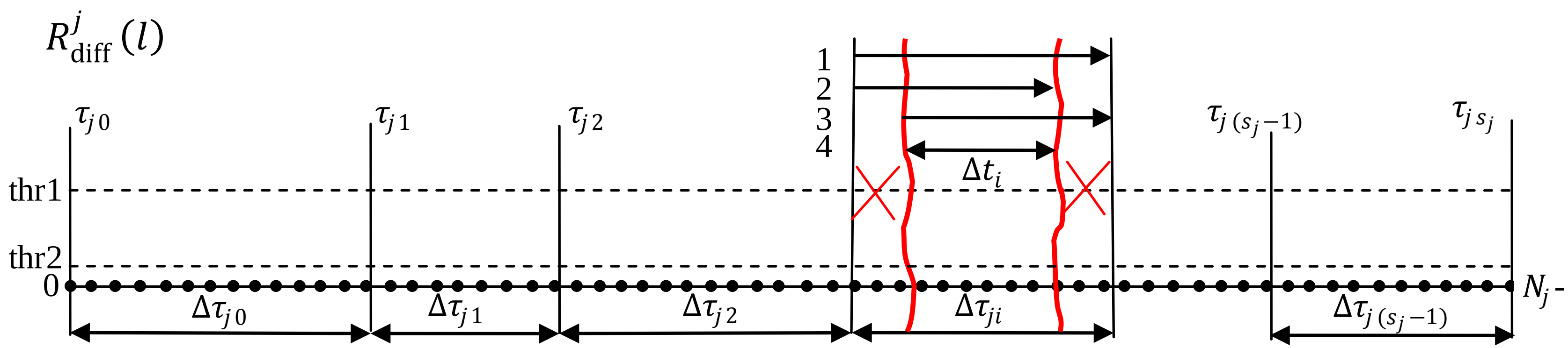
$\Delta\tau_{ji} = \tau_{j\,(i+1)} - \tau_{ji}$ is the length (in frame numbers) of i-th scene of the j-th Ad;

$i = (0, \dots, s_j - 1)$, where s_j is the number of scenes в j-th Ad.

The Ads video chains are: $R^{j}_{\text{chain}} = (\Delta\tau_{j\,0}, \Delta\tau_{j\,1}, \dots, \Delta\tau_{j\,(s_j-1)})$.

2.3. Detection of full and/or partial scenes from Adverts in TV

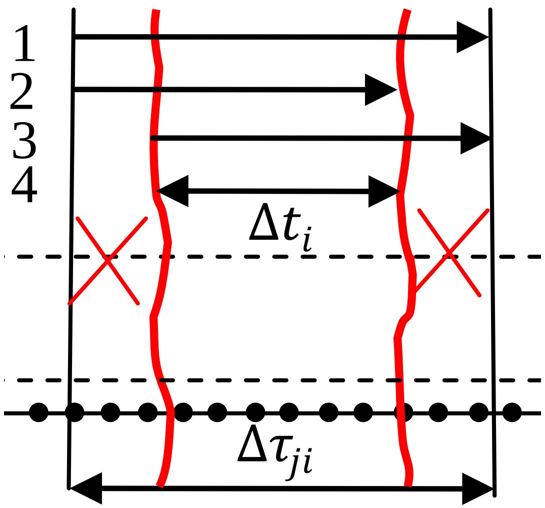
- Main rule: One scene from a TV video can be part of a scene in a given Ad (i.e. to match or to be shorter), but a scene from an Ad can only exactly to match with a scene from a TV video.



- The necessity of this rule arises because of the cases when we are given only the full Ads template, but not the shorter Ads variants, which consist of part(s) of the whole scenes from the full Ads.

2. Description of the proposed algorithm

2.3. Detection of full and/or partial scenes from Adverts in TV



Our **Algorithm** covers four cases of inclusion:

1. The scene $\Delta t_i \in$ TV exactly matches with the $\Delta \tau_{ji}$ scene from j-th Ad;
2. Only the beginning of the scene $\Delta t_i \in$ TV matches with the beginning of $\Delta \tau_{ji}$ scene;
3. Only the end of the scene $\Delta t_i \in$ TV matches with the end of $\Delta \tau_{ji}$ scene;
4. The beginning and the end of scene $\Delta t_i \in$ TV are internal for the $\Delta \tau_{ji}$ scene.

The **Algorithm** is applicable also for the cases when more than one scene Δt_i is internal for a given $\Delta \tau_{ji}$ scene.

2. Description of the proposed algorithm

2.3. Pseudo code of the algorithm

Algorithm:

for $\forall\, \Delta t_i \in \mathrm{I}_{\mathrm{chain}}$

  for $\forall\, j \in \mathcal{R}$

    if $\Delta\tau_{ji} \leq \Delta t_i$

      { if $\mathrm{SAD}(l, j) = \frac{1}{X \times Y} \sum_{\forall(x,y)} \left| \overline{I_{\Delta t_i}(x, y)} - \mathrm{I}_l^{\mathrm{j}}(x, y) \right| < \mathrm{thr2}$, for $\forall\, l \in \Delta\tau_{ji}$

        then

        scene $\Delta t_i \subseteq \Delta\tau_{ji}$ from j-th Ad is found in the TV stream

      }

  end

end

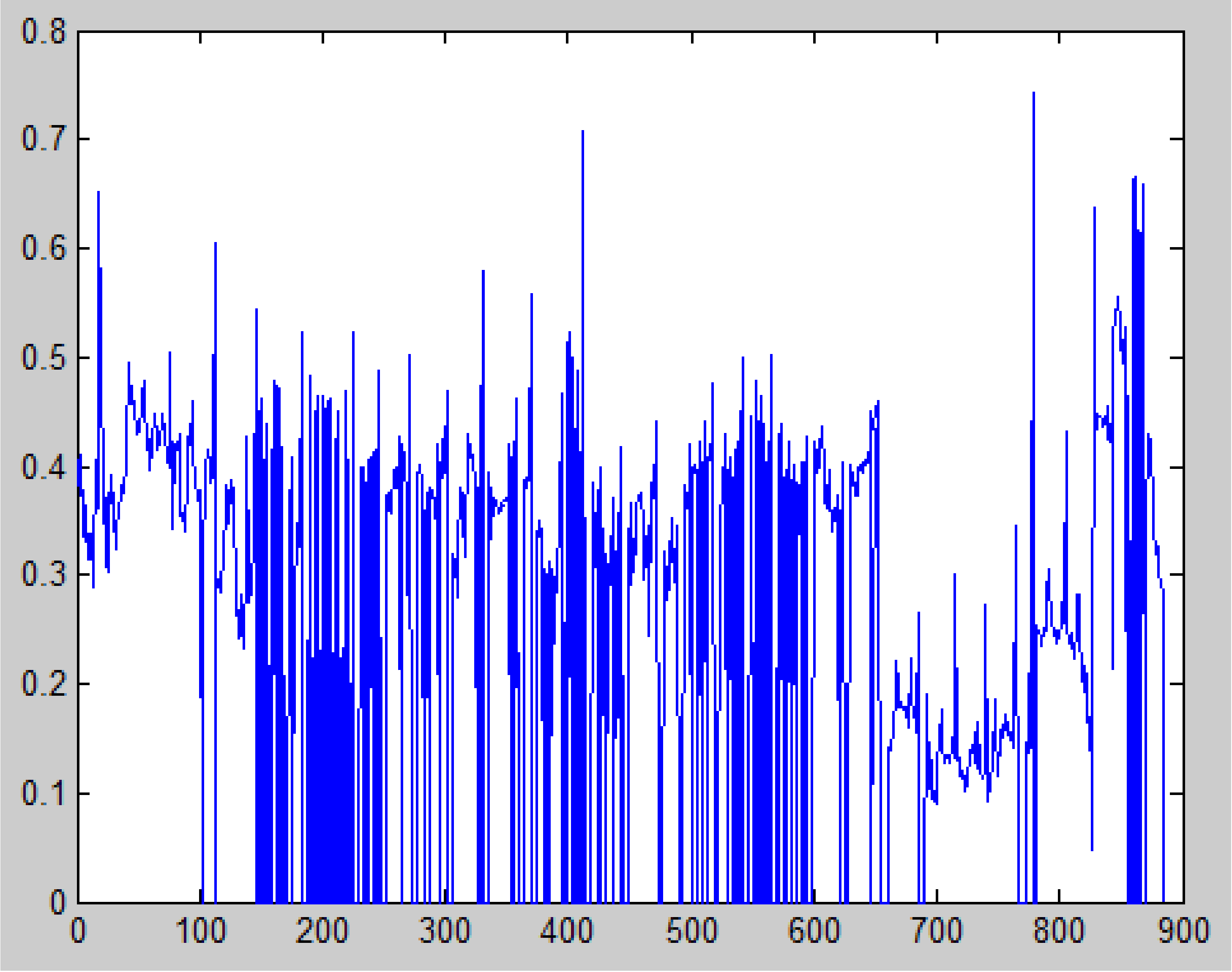
- thr2 is the optimal threshold for closeness of two visually equal frames, which are actually minimally different by content, because of the noise from the frames compression;
- $\overline{I_{\Delta t_i}(x, y)}$ is the average frame in the scene Δt_i from a TV video:

$$\overline{I_{\Delta t_i}(x, y)} = \mathrm{I}_{\frac{t_i + \Delta t_i - 1}{2}}(x, y)$$

Outline

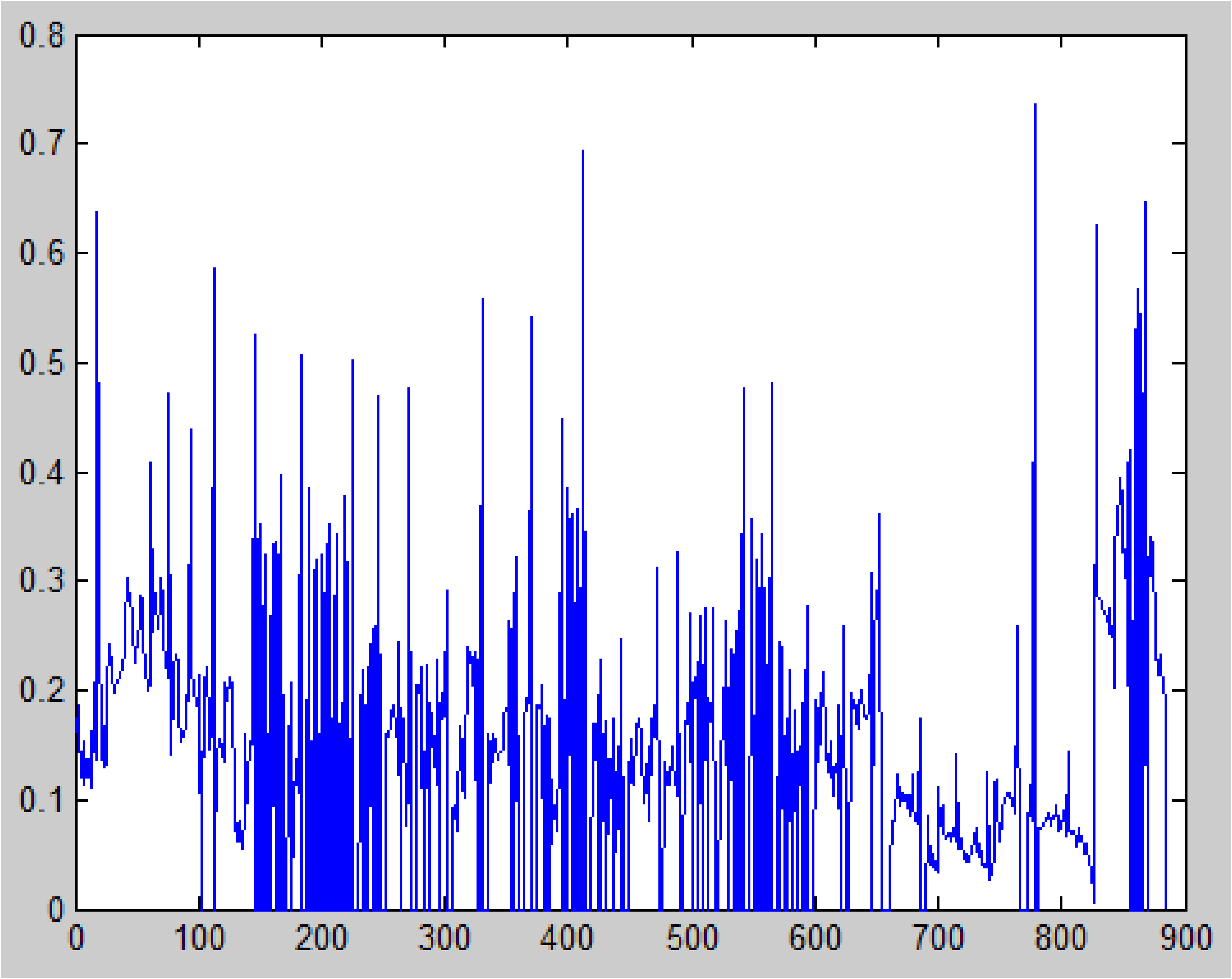
CROSS Online Bulgarian Network

3. Conducted experiments (pixel similarity thresh = 0)

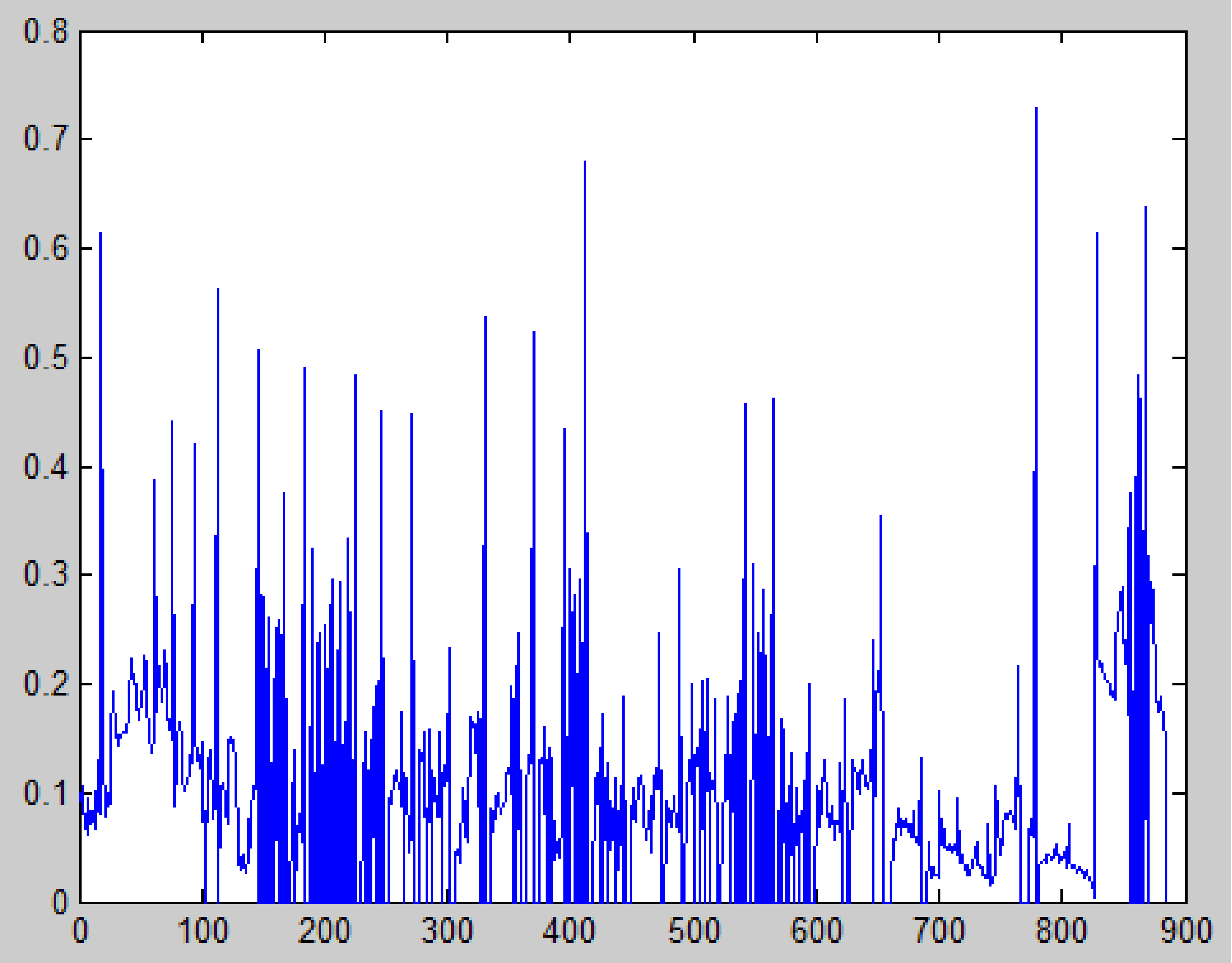


CROSS
Online Bulgarian Network

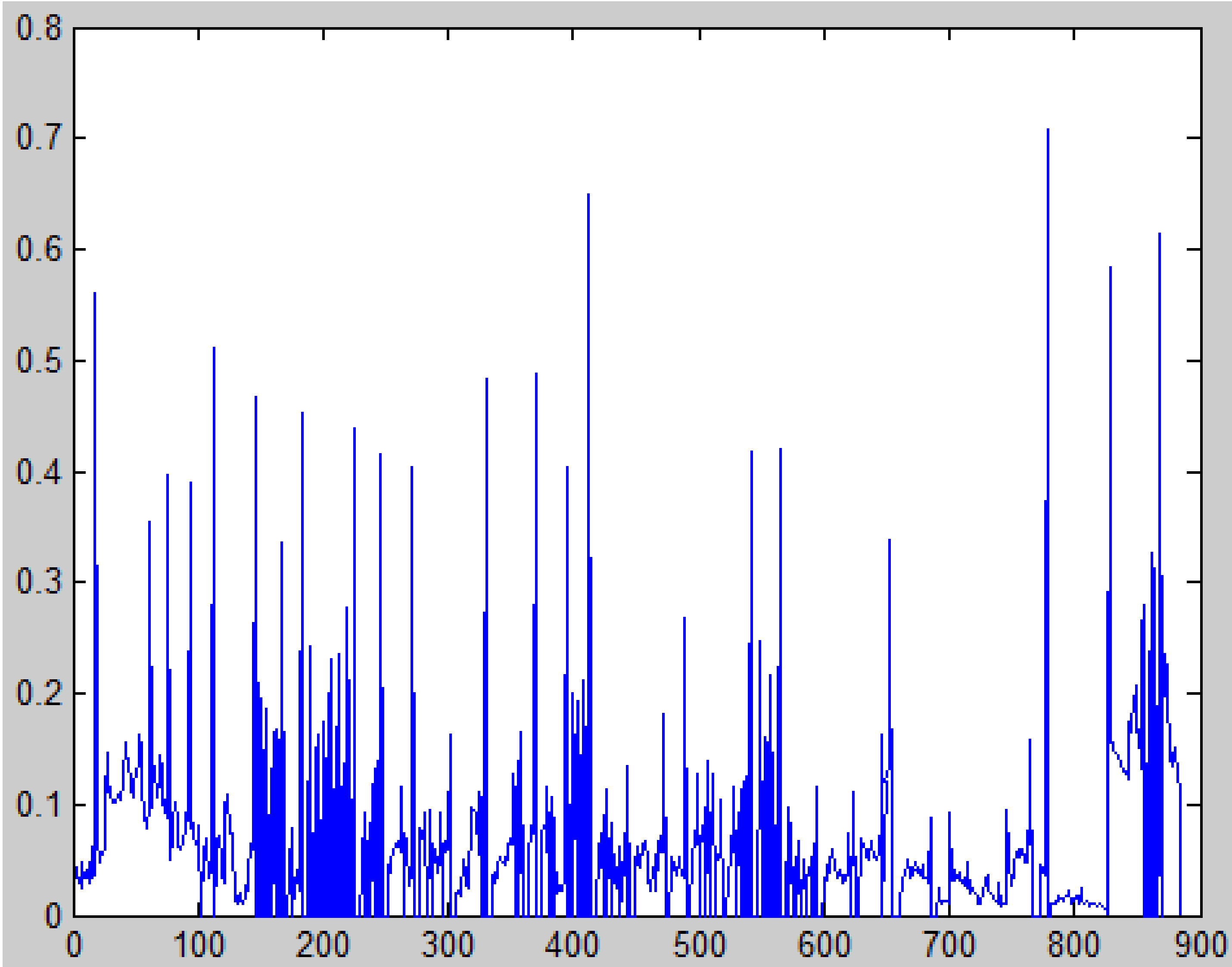
3. Conducted experiments (pixel similarity thresh = 4)



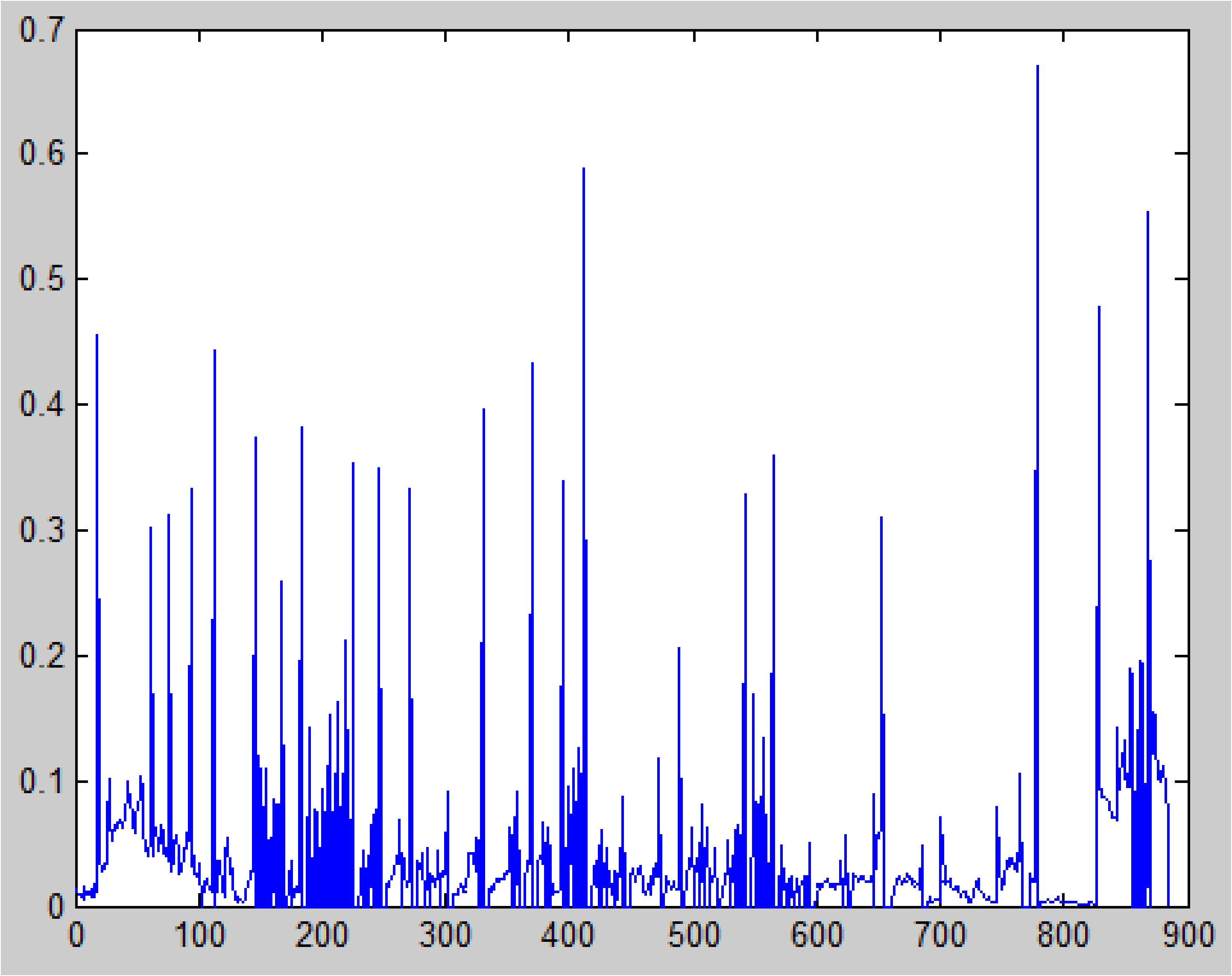
3. Conducted experiments (pixel similarity thresh = 8)



3. Conducted experiments (pixel similarity thresh = 16)

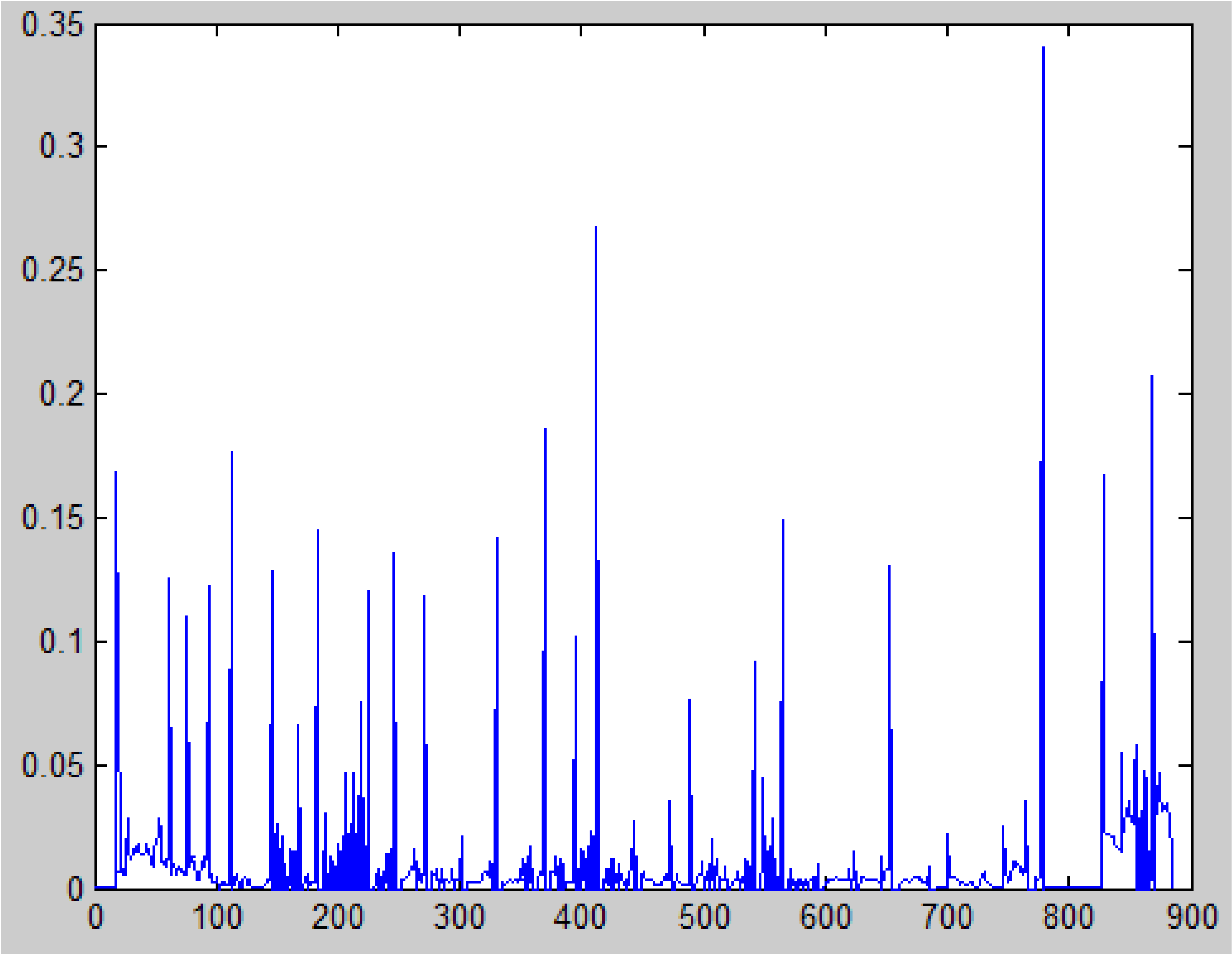


3. Conducted experiments (pixel similarity thresh = 32)

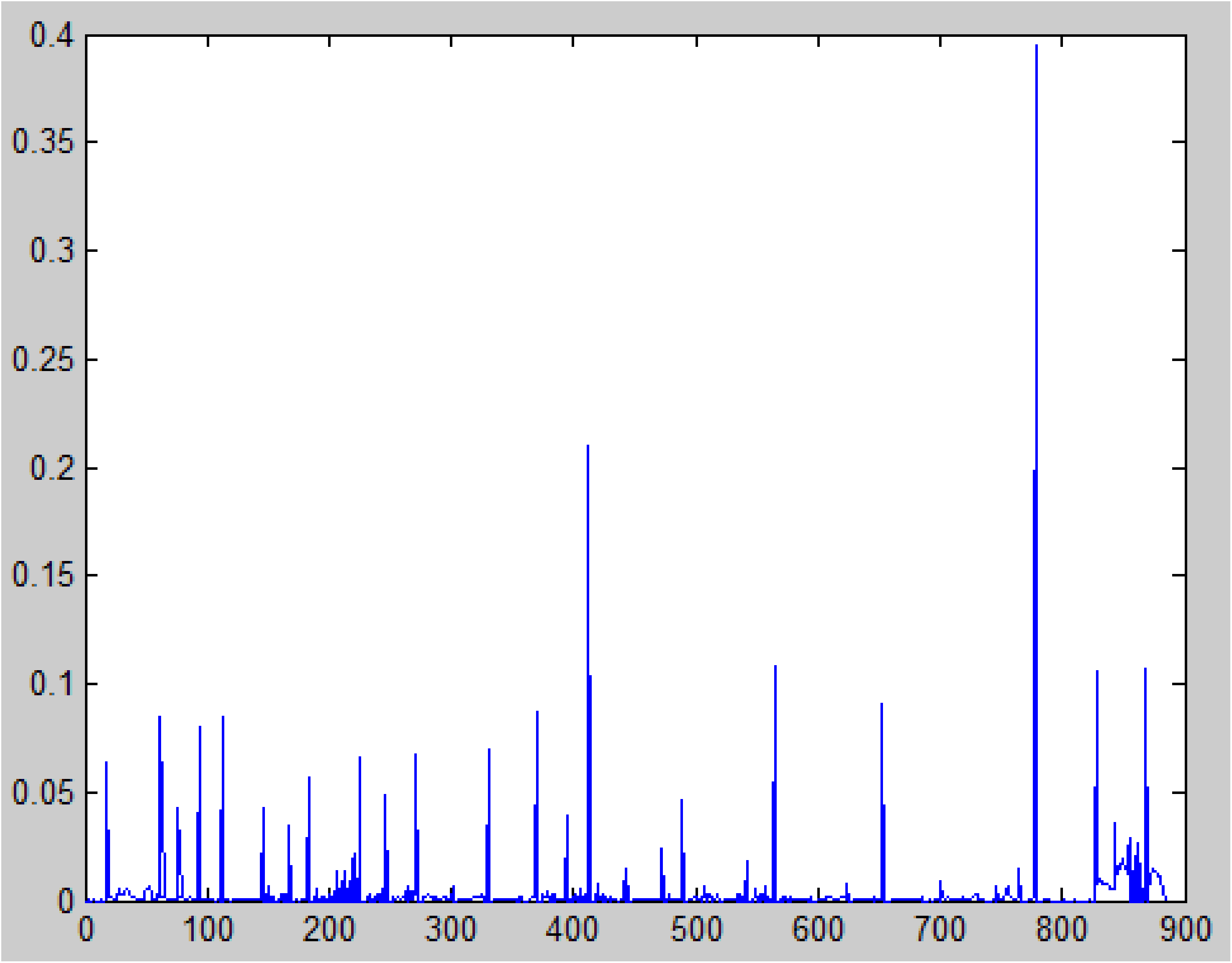


CROSS Online Bulgarian Network

3. Conducted experiments (pixel similarity thresh = 64)



3. Conducted experiments (pixel similarity thresh = 128)



CROSS Online Bulgarian Network

Outline

1. Problem formulation and the company's needs
   1.1. The simplest and the slowest intuitive solution
   1.2. Speed-up knowing JPEG's DCT coefficients in advance
   1.3. The fastest and entirely problem solving solution
2. Description of the proposed algorithm for full problem solving
   2.1. Forming differential videos (TV stream and advertisements)
   2.2. Scenes detection and videos representation as integer chains
   2.3. Detection of full and/or partial scenes from Adverts in TV
   2.4. Pseudo code of the proposed algorithm
3. Conducted experiments
4. **Conclusion words**

104th European Study Group with Industry
September 23 - 27, 2014, Sofia, Bulgaria

THANK YOU ☺

ESGI 104

CROSS Online Bulgarian Network

Dimo Dimov, IICT-BAS
Atanas Nikolov, IICT-BAS
Krassimira Ivanova, IMI-BAS
Maroussia Bojkova, IMI-BAS, FMI-SU
Plamen Mateev, IMI-BAS, FMI-SU
Ognian Kounchev, IMI-BAS
Vassil Kolev, IICT-BAS
Miroslav Ivanov, IMI-BAS