VIDEO SHOT BOUNDARY DETECTION VIA SEQUENTIAL CLUSTERING

Sergii Mashtalir, Volodymyr Mashtalir, Mykhailo Stolbovyi

Abstract: *To provide condensed and succinct representations of a video stream content there arise huge demand of video parsing. Key frames. shots, scenes, events, scenario represent structural video units. Primary and minimal unit is shot having similar visual contents and can be obtained by video stream clustering. Single-valued features of interest points neighborhoods associated with visual attention regions produce multidimensional time series. Detection of sharp and smooth boundaries of video shot based on fuzzy recurrent clustering is considered.*

Keywords: *I. Computing Methodologies, I.5. Pattern Recognition, I.5.3. Clustering*

Introduction

Large-scale video data collections growth at an exponential rate leads to development of novel on line approaches for structuring of source video streams automatically [Asghar et al, 2014]. Sequential semantic analysis of video structure is a key feature in video summarization, retrieval systems, etc. The peculiarity of video streams processing is on line data acquisition when frames get consistently with rather high frequency. But the foremost challenge when creating a video understanding system remains the semantic gap, i.e. disparity between high-level interpretation of a video and low-level features that can be extracted from frame sequences [Geetha and Narayanan, 2008]. Hierarchical video representation structure is described from the low level to high level: from frames to shots, scenes, events, scenario. The main video parsing units are shots having similar visual contents and can be obtained by video stream clustering. Shot boundary detection in temporal imagery is based usually on low-level frame feature changes such as in descriptions of interest points, boundaries, shapes, colors, textures, object motion, etc. [Kundu and Janwe, 2015]. Small neighborhoods of interest points, which have some identifiable property, have definite promise in video stream partition into elementary uninterrupted content units. In any case frame features may generate a multidimensional time series.

The change detection in properties of multidimensional time series (data sequences) is often encountered in many practical applications [Basseville and Nikiforov, 1993, Badavas, 1993, Pouliezos and and Stavrakakis, 1994]. However, most of the known on-line sequential detection algorithms are oriented to sudden-onset disruptions [Basseville and Nikiforov, 1993], while in video streams, changes can occur fairly smoothly by virtue of various gradual shot changes (dissolve, fade in, fade out, wipe, etc.) or just very slow panning. In such situations, methods of time series fuzzy segmentation [Abonyi et

al, 2003, Bodyanskiy and Mashtalir, 2012] are more preferable, however, due to their computational complexity, they become inefficient at high data entry rates for processing. Of course, one can use fuzzy clustering [Badavas, 1993, Aggarwal and Reddy, 2014] and, above all, methods of recurrent clustering [Park and Dagher, 1984, Bodyanskiy, 2005], which, however, suffer from the effects of the 'curse of dimensionality' (vectors in high dimensional spaces tend to have roughly the same length) generated by the relatively high dimensionality of the signals being processed. The approach based on the indirect time series clustering (problem is solved by defining a pairwise similarity or dissimilarity measure between series elements) can be very effective in combination with the methods of fuzzy recursive optimization [Hoeppner and Klawonn, 2000]. A technique of on line change properties detection of a multidimensional sequence subject to level clusters intersection is the study object of the paper.

Preprocessing of multidimensional time series for fuzzy clustering

Let original observations be given in the form of multidimensional sequence $x(1),\ldots,x(k),\ldots,x(N)$; $x(k)=(x_1(k),x_2(k),\ldots x_n(k))^T$ where $k=1,2,...,N,...$ denotes current discrete time. According to indirect approach to a time series clustering not the signal $x(k)$, $k=1,2...,N$ is partitioned into clusters The indirect clustering approach solves time series clustering problem by defining a pairwise similarity or by some indirect features such as average, variance, autocorrelation. So, for *i-th* series component, it is possible to introduce recurrent estimates:

– for an average

$$\bar{x}_i(k)=\bar{x}_i(k-1)+\frac{1}{k}\left(x_i(k)-\bar{x}_i(k-1)\right), \tag{1}$$

– for variance

$$\sigma_i^2(k)=\sigma_i^2(k)-\frac{1}{k}\left(x_i(k)-\bar{x}_i(k)^2-\sigma_i^2(k-1)\right)^2, \tag{2}$$

– for autocorrelation coefficients

$$r_i(k,\tau)=r_i(k-1,\tau)+\frac{1}{k}\left(\left(x_i(k)-\bar{x}_i(k)\right)\left(x_i(k-\tau)-\bar{x}_i(k)\right)-r_i(k-1,)\tau\right). \tag{3}$$

Denoting $\bar{x}(k)=(\bar{x}_1(k),\bar{x}_2(k)\ldots\bar{x}_n(k))^T$ and supposing that τ=0,1,2,…τ_{max} stands for time lag it is easy enough to identify vector-matrix counterparts of (1) – (3)

$$\bar{x}(k) = \bar{x}(k-1) - \frac{1}{k}\left(x(k) - \bar{x}(k-1)\right), \tag{4}$$

$$R(k,\tau) = R(k-1,\tau) + \frac{1}{k}\left(\left(x(k) - \bar{x}(k)\right)\left(x(k-\tau) - \bar{x}(k)\right)^T - R(k-1,\tau)\right) \tag{5}$$

where diagonal elements of a symmetric matrix of $R(k,0)$ are, in fact, estimations of variances $r_{ii}(k,0) = \sigma_i^2(k)$ and off-diagonal elements constitute coefficients of mutual correlation $r_{ij}(k,0)$, $i,j = 1,2,\ldots n$.

To provide adaptive properties of recurrent procedures (1)-(5), it seems desirable to estimate being studied characteristics on the basis of exponential smoothing modification [Pau, 1981]

$$\bar{x}_i^{\alpha}(k) = \alpha x_i(k) + (1-\alpha)\bar{x}_i^{\alpha}(k-1), \quad 0 < \alpha < 1, \tag{6}$$

$$(\sigma_i^{\alpha}(k))^2 = \alpha\left(x_i(k) + \bar{x}_i^{\alpha}(k)\right)^2 + (1-\alpha)\left(\sigma_i^{\alpha}(k-1)\right)^2, \tag{7}$$

$$r_i^{\alpha}(k,\tau) = \alpha\left(x_i(k) - \bar{x}_i^{\alpha}(k)\right)(x_i(k-\tau) - \bar{x}_i^{\alpha}(k)) + (1-\alpha)r_i^{\alpha}(k-1,\tau) \tag{8}$$

where $\alpha = 2/(L+1)$ is forgetting factor which provides smoothing at the sliding window containing the L last observations $x_i(k), x_i(k-1), \ldots x_i(k-L+1)$.

Thus, $(2+\tau_{max})\times 1$ feature vector $\tilde{x}_i(k) = \left(\bar{x}_i^{\alpha}(k),\ (\sigma_i^{\alpha}(k))^2,\ r_i^{\alpha}(k,1)\ldots r_i^{\alpha}(k,\tau_{max})\right)^T$ can be stated in correspondence to each time series component $x_i(k)$ and namely this vector variety represents clustering objects.

To process $(n+1)$-dimensional signal for an average vector $\bar{x}^{\alpha}(k)$ and an autocorrelation matrix $R^{\alpha}(k,\tau)$ evaluations it is also possible to use the procedure of exponential smoothing in the form [Bodyanskiy et al, 2012]

$$\bar{x}^{\alpha}(k) = A\,x(k) + (I-A)\bar{x}^{\alpha}(k-1) \tag{9}$$

where $A = diag\,(\alpha_1, \alpha_2, \ldots \alpha_n)$, I denotes $(n \times n)$ identity matrix,

$$R_i^{\alpha}(k,) = A\left(\left(x(k) - \overline{x}^{\alpha}(k)\right)\left(x(k-\tau) - \overline{x}_i^{\alpha}(k)\right)^T\right)_i + (I - A)R_i^{\alpha}(k) \tag{10}$$

where index $i = 1,2,\ldots n$, stands for column number of corresponding matrix.

Formulae (9), (10) application leads to necessity of $0.5\left(n^2 + n\right)\left(\sigma_{max} + 1\right) + n$ parameters persistently parsing what considerably complicates real time data processing. In connection with requirements of processing speed increasing, further it is offered for study and, respectively, on-line clustering to use the current values of $((2 + \tau_{max}) \times n)$ matrix $\tilde{x}(\mathrm{k}) = \left(\tilde{x}_1(k), \widetilde{x}_2(k), \ldots \tilde{x}_n(k)\right)$ whose elements are refined at each step by means of (6)-(8) relationships.

Change detection of time series components properties

Since changes, occurring during multidimensional signal $\tilde{x}(k)$ processing, can be both jump-like and smooth, fuzzy clustering algorithms (with different fuzzifiers) based on the fuzzy goal functions usage can be applied [Aggarwal and Reddy, 2014].

Now, let formalize a problem. Let an analysis object be the i-th component of the $(\alpha + \tau_{max}) \times 1$ vector signal $\tilde{x}_i(k) = \left(\overline{x}_i^{\alpha}(k),\ (\sigma_i^{\alpha}(k))^2,\ r_i^{\alpha}(k,1) \ldots r_i^{\alpha}(k, \tau_{max})\right)^T$ and the traditional self-learning criterion is used as the objective function

$$E\left(\mu\left(\tilde{x}_i(k), C_i(l)\right)\right) = \sum_{k=1}^{N}\sum_{l=1}^{m} \mu^{\beta}\left(\tilde{x}_i(k), C_i(l)\right) D^2\left(\tilde{x}_i(k), C_i(l)\right) \tag{11}$$

under constraints

$$\sum_{l=1}^{M} \mu\left(\tilde{x}_i(k), C_i(l)\right) = 1, \quad \forall k = 1,2,\ldots N, \tag{12}$$

$$0 < \sum_{k=1}^{N} \mu\left(\tilde{x}_i(k), C_i(l)\right) \le N, \quad \forall l = 1,2,\ldots,m. \tag{13}$$

Here, $\mu\left(\tilde{x}_i(k), C_i(l)\right)$ is the membership level of the vector $\tilde{x}_i(k)$ to the l-th cluster, $C_i(l)$ is the centroid of this cluster to be evaluated, β is a nonnegative parameter called the fuzzifier and controlled how much clusters may overlap, $D^2\left(\tilde{x}_i(k), C_i(l)\right)$ denotes some measure specifying the distance between the vectors $\tilde{x}_i(k)$ and cluster $C_i(l)$ in the accepted metric (usually Euclidean in the simplest and most common case). In this case, the objective function (11) can be rewritten in the form

$$E\left(\mu\left(\tilde{x}_i(k), C_i(l)\right)\right) = \sum_{k=1}^{N}\sum_{l=1}^{m}\mu^2\left(\tilde{x}_i(k), C_i(l)\right) \| \tilde{x}_i(k) - C_i(l) \|^2. \tag{14}$$

Optimization of objective function (11) by means of the standard nonlinear programming technique leads to the result known as fuzzy C-means clustering procedure

$$\begin{cases} \mu\left(\tilde{x}_i(k), C_i(l)\right) = \dfrac{\| x_i(k) - C_i(l) \|^{-2}}{\sum_{j=1}^{m} \| \tilde{x}_i(k) - C_i(j) \|^{-2}}, \\ C_i(l) = \dfrac{\sum_{k=1}^{N} \mu^2\left(\tilde{x}_i(k), C_i(l)\right) \tilde{x}_i(k)}{\sum_{k=1}^{N} \mu^2\left(\tilde{x}_i(k), C_i(l)\right)}. \end{cases} \tag{15}$$

It is easily seen that method (15) describes a batch procedure for information processing, when the entire sample to be analyzed is given and does not change during operations. It is clear that such a procedure can not be used to detect changes under sequential video processing. To achieve specific aims of on line video parsing it will be sufficient to use the adaptive version of (15) [Bodyanskiy, 2005], which allows real-time solving the clustering-segmentation problem as data are fed for processing. In the above notation, the clustering procedure can be written in more promising form

$$\begin{cases} \mu\left(\tilde{x}_i(k), C_i(I,k-1)\right) = \dfrac{\| \tilde{x}_i(k) - C_i(I,k-1) \|^{-2}}{\sum_{j=1}^{m} \| \tilde{x}_i(k) - C_i(j,k-1) \|^{-2}}, \\ C_i(I,k) = C_i(I,k-1) + \eta(k)\, \mu^{\beta}\left(\tilde{x}_i(k), C_i(I,k-1)\right)\left(\tilde{x}_i(k) - C_i(I,k-1)\right) \end{cases} \quad (16)$$

where $0 < \eta(k) < 1$ is the learning rate parameter, chosen usually from empirical considerations.

With $\beta = 2$ procedure (16) will lead to the same results as the batch algorithm (15), but at the same time it allows to process a data sequence that is received in real time for processing. When $\beta = 0$ (16) takes the form of Kohonen's self-learning algorithm [Kohonen, 1995], which is one of the most popular in solving clustering problems. Note also that conditions $\beta = 0$, $\eta(k) = k^{-1}$ correspond to the recurrent version of the *k* -means clustering [Aggarwal and Reddy, 2014], widely used in video and image processing. Finally, an important advantage (17) is that preconditions are created for filtering some local distortions, in particular, occlusions.

Change properties detection of a multidimensional sequence

When segmenting video streams with sequential mode of one-dimensional discrepancies detection, an application of usual pixels as a time series generator is not valid. Explanation is simple: in the frame sequence for such processing it is possible to use only the gray level (color) values at given coordinates. In other words, the spatial content of the image is completely excluded from consideration that increases the semantic gap. In addition, the impact of various disturbing influences, primarily distorting the coordinates, hides natural temporal dynamics of the point to be analyzed. Thus, it is necessary to consider point neighborhoods properties producing time series. For sequential video clustering in order to reduce the semantic gap, it is most expedient to select, as point-generators of time series, interest points which are sufficiently uniformly distributed in the field of view and preferably are inside visual attention regions what further increases the interpretability of the detected discrepancies.

Interest points (corners, contour junctions, ridges extremum, etc.) utilizing may provide feature point correspondences between frames from video stream and, ipso facto, generate multidimensional time series corresponding to video content changes with high degree of robustness since they have distinctness, well-defined position, local information contents, are stable under illumination/brightness variations. Thus, shot boundary detection may be based on single-valued textures or another features defined in neighborhoods of various interest points.

Since in the detection process described above, *n* one-dimensional type (16) procedures are

simultaneously realized, from the computational point of view it is more efficient to process at once parameters of the entire matrix $\tilde{x}(k)$ with dimensionality $((2+\tau_{max})\times n)$. To do this, it is advisable to use the fuzzy C-means technique modification where instead of objective function (14) it can be exploited expression

$$E(\mu(\tilde{x}(k),C(l))=\sum_{k=1}^{N}\sum_{l=1}^{m}\mu^2(\tilde{x}(k),C(l))Sp(\tilde{x}(k)-C(l))(\tilde{x}(k)-C(l))^T. \quad (17)$$

Here Frobenius norm metric is applied instead of the Euclidean one and, in addition, m matrix-centroids $C(l)$ are introduced.

Minimization of (17) with constraints (12), (13) leads to analytical expression

$$\begin{cases} \mu(\tilde{x}(k),C(l))=\dfrac{\left(Sp(\tilde{x}(k)-C(l))(\tilde{x}(k)-C(l))^T\right)^{-1}}{\sum_{j=1}^{m}\left(Sp(\tilde{x}(k)-C(j))(\tilde{x}(k)-C(j))^T\right)^{-1}}, \\ C(l)=\dfrac{\sum_{k=1}^{N}\mu^2(\tilde{x}(k),C(l))\tilde{x}(k)}{\sum_{k=1}^{N}\mu^2(\tilde{x}(k),C(l))}. \end{cases} \quad (18)$$

Obviously, relationships (18) can not be utilized to solve on-line being studied problem. In this case, it is expedient to use the matrix modification of the recurrent procedure (16), which in this case has the form

$$\begin{cases} \mu(\tilde{x}(k),C(l,k-1))=\dfrac{\left(Sp(\tilde{x}(k)-C(l,k-1))(\tilde{x}(k)-C(l,k-1))^T\right)^{-1}}{\sum_{j=1}^{m}\left(Sp(\tilde{x}(k)-C(j,k-1))(\tilde{x}(k)-C(j,k-1))^T\right)^{-1}}, \\ C(l,k)=C(l,k-1)+\eta(k)\mu^{\beta}(\tilde{x}(k),C(l,k-1))(\tilde{x}(k)-C(l,k-1)) \end{cases} \quad (19)$$

where the learning rate parameter $\eta(k)$ is chosen from the same considerations as in (16).

It can also be noted that when $\beta = 0$ (19) takes the form of a clear matrix self-learning algorithm for on-line clustering problem solutions as follows

$$C(l,k) = C(l,k-1) + \eta(k)\left(\tilde{x}(k) - C(l,k-1)\right)$$

which is, in fact, a matrix version of the of self-learning rule of T. Kohonen principle 'Winner Takes All'.

Conclusion

Any real problem of video understanding is extremely difficult due to necessity of valid on line parsing of enormous image sequences in feature or signal spaces. Though video shot boundary detection techniques have achieved significant progress, features to be utilized and reflect desirable semantic level remain an open issue. One from promising approaches is to use interest points associated with visual attention regions, more precisely, the local neighborhoods of these interest points. In this case semantic gap may be decreased and sufficiently high-speed recurrent procedures can be achieved. The problem of change detection in multidimensional (matrix) data streams getting for processing in real time is considered. The on-line procedures for fuzzy clustering have been introduced. It makes possible on line clustering for both slow and abrupt changes. The proposed approach is fairly simple and can find different application, first of all, when processing video streams.

Acknowledgment

The paper is published with partial support by the ITHEA ISS (www.ithea.org) and the ADUIS (www.aduis.com.ua)

Bibliography

[Abonyi et al, 2003] Abonyi J., Feil B., Nemett S., Arva P., Fuzzy clustering based segmentation of time-series. Lecture Notes in Computer Science. 2810. Berlin: Springer, 2003. pp. 275–285. ISBN 978-3-540-40813-0 (print version), ISBN 978-3-540-45231-7 (on-line), DOI: 10.1007/978-3-540-45231-7_26 http://link.springer.com/chapter/10.1007/978-3-540-45231-7_26.

[Aggarwal and Reddy, 2014] Aggarwal C.C., Reddy C.K. Data clustering: Algorithms and applications. Boca Raton: CRC Press, 2014.

[Asghar et al, 2014] Asghar M.N., Hussain F., Manton R., Video indexing: a survey. International Journal of Computer and Information Technology, Vol. 3, Issue 1. 2014. pp. 148-169. ISSN: 2279-0764, http://www.ijcit.com/archives/volume3/issue1/Paper030123.pdf

[Badavas, 1993] Badavas P.C. Real-time statistical process control. Englewood Cliffs: Prentice-Hall, 1993.

[Basseville and Nikiforov, 1993] Basseville M., Nikiforov I. Detection of abrupt changes: Theory and application. Englewood Cliffs, N.J.: PTR Prentice-Hall, 1993.

[Bodyanskiy and Mashtalir, 2012] Bodyanskiy Ye.V., Mashtalir S.V., Search for video stream changes via multidimensional time series analysis. Reports of the National Academy of Sciences of Ukraine, No11, 2012. pp. 30-33, ISSN 1025-6415 (print version), ISSN 2518-153X (on-line), https://dopovidi.nas.gov.ua/2012-11/12-11-05.pdf

[Bodyanskiy et al, 2012] Bodyanskiy Ye., Kinoshenko, D., Mashtalir, S., Mikhnova, O., On-line video segmentation using methods of fault detection in multidimensional time sequences. International Journal of Electronic Commerce Studies, Vol. 3, Issue 1. 2012. pp. 1-20. ISSN 2073-9729, http://academic-pub.org/ojs/index.php/ijecs/article/download/1010/122&authuser=1&usg=AFQjCNEhzFr7iL5POexC5yhRmCiP -pVeQg&sig2=tAh3O9zVY5KXPk5fNgPC-g&bvm=bv.150729734,d.bGg.

[Bodyanskiy, 2005] Bodyanskiy Ye., Computational intelligence techniques for data analysis. Lecture Notes in Informatics, Vol. 72. Bonn: GI, 2005. pp. 15-36. ISBN 3-88579-401-2 http://subs.emis.de/LNI/Proceedings/Proceedings72/GI-Proceedings.72-1.pdf

[Geetha and Narayanan, 2008] Geetha P.A., Narayanan V., Survey of content-based video retrieval. Journal of Computer Science, Vol. 4, Issue 6. 2008. pp. 474-486. ISSN 1549-3636, DOI: 10.3844/jcssp.2008.474.486, http://thescipub.com/PDF/jcssp.2008.474.486.pdf.

[Hoeppner and Klawonn, 2000] Hoeppner, F., Klawonn, F., Fuzzy clustering of sampled functions. In: Proc. 19th Int. Conf. of the North American Fuzzy Information Processing Society (NAFIPS), Atlanta, USA. 2000. pp. 251 – 255. ISBN: 0-7803-6274-8 (print version), DOI: 10.1109/NAFIPS.2000.877431, http://public.fh-wolfenbuettel.de/~hoeppnef/paper/Hoeppner-NAFIPS-2000.pdf

[Kohonen, 1995] Kohonen, T. Self-Organizing Maps. Berlin: Springer-Verlag, 1995.

[Kundu and Janwe, 2015] Kundu A., Janwe N., A survey on video segmentation the future roadmap. International Journal of Modern Trends in Engineering and Research, Vol. 2, Issue 3. 2015. pp. 527-534. ISSN: 2349-9745 (on-line), ISSN: 2393-8161 (print version), http://www.academia.edu/13545611/ a_survey_on_video_segmentation_the_future_roadmap

[Park and Dagher, 1984] Park D.C., Dagher I., Gradient based fuzzy c-means (GBFCM) algorithm. In: IEEE Int. Conf. on Neural Networks, 1984. pp. 16.26-16.31. ISBN: 0-7803-1901-X (print version), DOI: 10.1109/ICNN.1994.374399, http://ieeexplore.ieee.org/document/374399/

[Pau, 1981] Pau L.F. Failure Diagnosis and performance monitoring. NY: Marcel Dekker Inc., 1981.

[Pouliezos and and Stavrakakis, 1994] Pouliezos A.D., Stavrakakis G.S. Real-time fault monitoring of industrial processes. Dordrecht: Kluwer Academic Publishers, 1994.

Authors' Information

Mashtalir Sergii *– Kharkiv National University of Radio Electronics, ass. professor of Informatics department. 14, Nauky Ave., Kharkiv, 61166, Ukraine;*

e-mail: sergii.mashtalir@nure.ua

Major Fields of Scientific Research: Temporal video streams processing, image processing

Mashtalir Volodymyr *– Kharkiv National University of Radio Electronics, professor of Informatics department. 14, Nauky Ave., Kharkiv, 61166, Ukraine;*

e-mail: volodymyri.mashtalir@nure.ua

Major Fields of Scientific Research: Pattern recognition, image processing

Stolbovyi Mykhailo *– Kharkiv National University of Radio Electronics, PhD student. 14, Nauky Ave., Kharkiv, 61166, Ukraine*

Major Fields of Scientific Research: Pattern recognition, image and video processing