Educational video classification by using a transcript to image transform and supervised learning

Houssem Chatbri[1], Marlon Oliveira[2], Kevin McGuinness[1], Suzanne Little[1], Keisuke Kameyama[3], Paul Kwan[4], Alistair Sutherland[2] and Noel E. O'Connor[1]
[1] Insight Centre for Data Analytics, Dublin City University, Ireland
e-mail: houssem.chatbri@dcu.ie, kevin.mcguinness@dcu.ie, suzanne.little@dcu.ie, noel.oconnor@dcu.ie
[2] School of Computing, Dublin City University, Ireland
e-mail: marlon.oliveira2@mail.dcu.ie, alistair.sutherland@dcu.ie
[3] Faculty of Engineering, Information and Systems, University of Tsukuba, Japan
e-mail: keisuke.kameyama@cs.tsukuba.ac.jp
[4] School of Science & Technology, University of New England, NSW, Australia
e-mail: paul.kwan@une.edu.au

***Abstract*— In this work, we present a method for automatic topic classification of educational videos using a speech transcript transform. Our method works as follows: First, speech recognition is used to generate video transcripts. Then, the transcripts are converted into images using a statistical co-occurrence transformation that we designed. Finally, a classifier is used to produce video category labels for a transcript image input. For our classifiers, we report results using a convolutional neural network (CNN) and a principal component analysis (PCA) model. In order to evaluate our method, we used the Khan Academy on a Stick dataset that contains 2,545 videos, where each video is labeled with one or two of 13 categories. Experiments show that our method is effective and strongly competitive against other supervised learning-based methods.**

***Keywords*— Educational video classification, transcript features, convolutional neural networks (CNN), principal component analysis (PCA)**

I. Introduction

In recent years, a growing number of highly-regarded academic institutions adopted massive open online courses (MOOC) and started to collaborate with online platforms such as Coursera, Udemy, and Khan Academy [21]. This led to a big interest by students worldwide and has been translated into an emerging online industry that formed large communities of MOOC students and educators [5].

Educational videos are now produced frequently. Therefore, systems for their storage, indexing, classification, and retrieval should be designed and constantly improved to maintain the sustainability of MOOC platforms. In this paper, we focus on automatic video classification which aims to classify a video into one or many of known categories that describe the video content. Automatic video classification is of important benefit for numerous applications of video analysis such as content-based retrieval and content recommendation.

We propose an approach that exploits an intrinsic characteristic of educational MOOC videos, which is the fact that the video topic category can be extracted from the speech. For this purpose, our approach extracts the video transcript in a first step, and uses a supervised learning model for classification as a last step. In between, transcripts are transformed into images and the problem is turned into visual pattern recognition. The transcript to image conversion is done via a statistical co-occurrence transform that we designed.

978-1-5386-1842-4/17/$31.00

For our experiments, we use a dataset from the Khan Academy educational platform, which has become one of the most popular MOOC platforms due to the high quality and intuitiveness of its videos [26]. We compare our approach with several other methods and we report significantly improved performances.

The remainder of this paper is as follow: In Sec. II, we overview key automatic video classification methods. Sec. III presents the Khan Academy video dataset that we used. Sec. IV describes our video classification method. We report our results in Sec. V, and finally our concluding remarks and future directions in Sec. VI.

II. Related work

The literature on automatic video classification is wide and it includes research on action recognition [11], anomaly detection [18], lecture video classification [3], etc. Video classification methods have traditionally used shallow features [4] before deep learning based approaches started to emerge [29].

Shallow features correspond to hand-crafted visual descriptors that encode appearance and motion information. This includes the Histogram of Oriented Gradients (HOG) [7], Histogram of Optical Flow (HOF) [6], and spatio-temporal interest points [16]. Local feature descriptors are extracted using dense grids [27] or by interest point detection [15], [16], and graph structures have been used to encode spatio-temporal information [10]. On the other hand, video classification has been achieved by using text features. This includes video closed captions and viewable text (e.g. scene text, news bar), in which case OCR is used to extract text from video frames [4], in addition to features that are extracted from video transcripts [3].

Table 1: Topics of the Khan Academy on a Stick dataset videos.

Category	Label
Math	Algebra, Calculus, Geometry, Trigonometry, Arithmetic, Differential Equations, Probability
Science	Biology, Cosmology and Astronomy, Organic Chemistry, Chemistry, Healthcare and Medicine, Physics

Contrary to shallow methods, deep learning techniques automatically learn discriminant features for video classification and they leverage the abundance of large amount of online videos [1]. Convolutional Neural Networks (CNNs) have been used for this purpose and they are fed with single frames or stacked frames [11]. Deep Learning approaches are suitable to data with discriminative visual information, so they perform well in datasets where object motion is an important feature such as human action datasets [24]. Research has also been conducted to deal with noisy data [2].

III. The Khan Academy video dataset

Massive open online courses (MOOC) received a lot of interest and become a commodity for a large number of students worldwide [5]. The concept has grown from recording lectures or talks and providing them online (such as in VideoLectures.net) to producing high quality educational videos that follow a specific template.Khan Academy is one of the most popular MOOC platform and it has gained large popularity among students in recent years due to the high quality of its videos and the excellent presentation skills, which led to its integration in a number of educational institutions [19], [26]. Usually, Khan Academy videos contain freehand sketched content on a digital tablet (Fig. 1).

In order to help institutions in developing countries with limited Internet access, Khan Academy put together an offline dataset called the *Khan Academy on a Stick* dataset[1]. This dataset contains 2,545 videos that were recorded between 2006 and 2013. The videos depict sketched content on a black background (Fig. 1), and they are annotated with 13 labels (Table 1). Fig. 2 shows the histogram of video labels, Fig. 3 shows the histogram of video durations, and Fig. 4 shows the histogram of video frame resolutions. It can be seen from the figures that the distributions of classes, video durations and frame resolutions are not balanced. Of the 2,545 videos, 238 have more than one label (e.g. Algebra and Trigonometry, Biology and Healthcare and Medicine).

Although Khan Academy provides a rich repository of videos for free, not much research was reported from the computational intelligence community. To the best of our knowledge, the work by Shin et al. on generating *visual transcripts* (i.e. structured visual documents) from Khan Academy videos [23] is worth noting and shares the same background with our work.

[1] http://khan.mujica.org

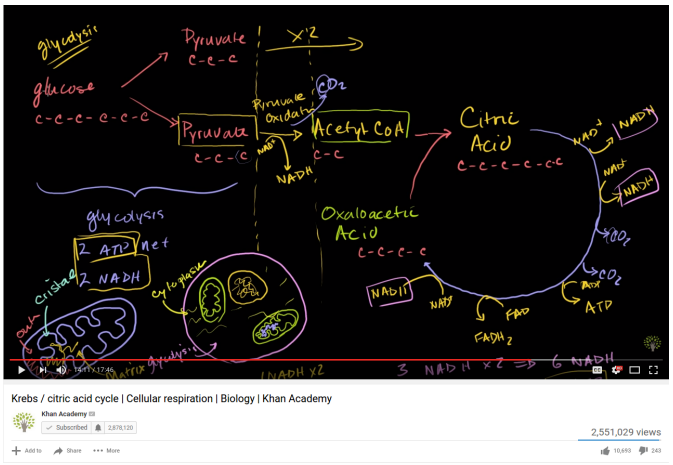

Fig. 1: A Khan Academy video with black background and colored sketched content, which is the style used in the Khan Academy on a Stick dataset.

IV. Automatic video classification using transcript to image transform

In this work, we rely on text features instead of frame visual features. We generate the video transcripts using a standard toolkit (Sec. IV). Then, we split them into 80% for training and 20% for testing while making sure that all video labels and label combinations are represented in that ratio.

We rely on transcripts instead of the frames' visual information for the following reasons:

- The speech contains keywords that are associated with certain topics, and they are usually easily discriminated. Contrarily, frames tend to follow the same template by the video producer, which makes them harder to discriminate video topics.
- Visually similar sketches are used in videos of different topic labels, e.g. Calculus and Arithmetic, Chemistry and Organic Chemistry, etc. This makes discrimination based on visual features more challenging.

Our approach starts by extracting the transcript of each video using the CMU Sphinx toolkit [14]. This tool is based on Hidden Markov Models (HMM) and it reaches word error rates (WER) of 26.9% and 22.7% on the VM1 and WSJ1 datasets respectively [8]. Afterwards, transcripts are converted into images using a co-occurrence transform that we designed (Sec. IV-A). Finally, the transcript image is fed to a CNN that produces the labels of the video (Sec. IV-B.1).

A. Transcript to image transform

The purpose of this step is to convert each transcript to an image-like representation that characterizes the transcript file content and that can be fed to a CNN classifier. To this end, we designed a statistical co-occurrence transform that works as follows (Algorithm 1): Considering a transcript T as an array of characters, each five adjacent characters

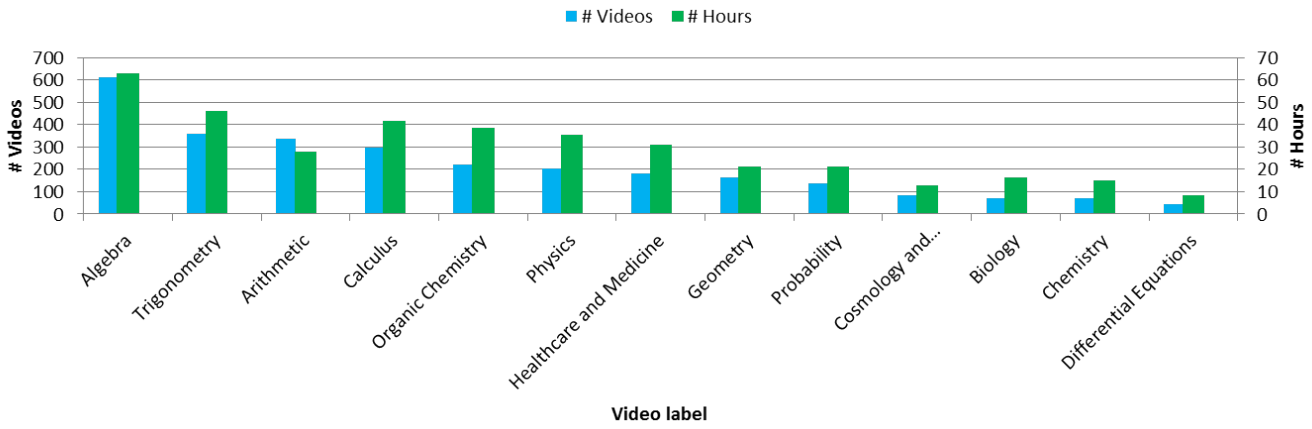

Fig. 2: Number of videos and footage duration for each video label.

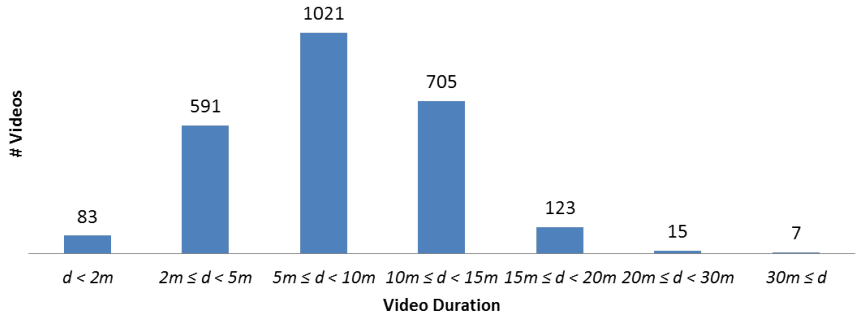

Fig. 3: Number of video for each duration interval.

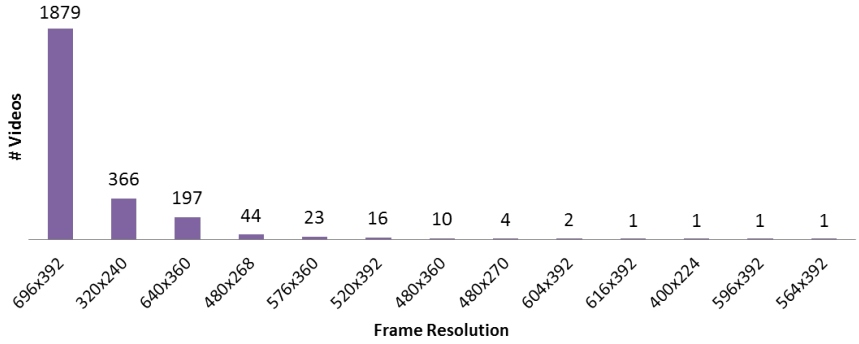

Fig. 4: Number of video of each frame resolution.

$C = T[j \ mod \ length(\mathrm{T})], j = i, ..., i{+}4\}$ are used to populate a 128×128 image I by using the ASCII codes of the first four characters of C to calculate a pixel coordinate and the ASCII code of the fifth character as an increment to the existing pixel value (128 corresponds to the total number of ASCII codes). Finally, I is normalized by dividing on its maximum cell value.

The result of the proposed transform can be visualized with grayscale images (Fig. 5). Despite of their sparseness, we expect high performances by a CNN classifier due to their proven effectiveness when coupled with sparse encoded features [28].

B. Classification models

1) A convolutional neural network (CNN) based approach: We use a Convolutional Neural Network (CNN) to produce the labels of a video transcript image from labels among the 13 video topics (Table 1). As illustrated in Fig. 6, our CNN has a Zero Padding layer to make sure that all transcript image pixels are considered, 3 convolutional layers with ReLU non-linearity, 1 fully connected layer with 128 neurons and

Algorithm 1 Transcript to image transform

Precondition: Transcript file T : array of characters

function TRANSCRIPTTOIMAGE(T)
  define I : 128×128 matrix
  for $i \leftarrow 1$ to $length(T)$ **do**
    $C \leftarrow \{T[j \ mod \ length(\mathrm{T})], j = i, ..., i+4\}$
    $x \leftarrow$ | ASCII ($C[1]$) - ASCII ($C[0]$) |
    $y \leftarrow$ | ASCII ($C[3]$) - ASCII ($C[2]$) |
    $v \leftarrow$ ASCII ($C[4]$)
    $I[x, y] \leftarrow I[x, y] \ + \ v$
  end for
  $I \leftarrow \frac{1}{max(I)} \ I$
  return I
end function

Fig. 5: Examples of transcript images (pixel values are normalized by $\times 255$). The left image corresponds to a biology video, and the right one corresponds to a physics video.

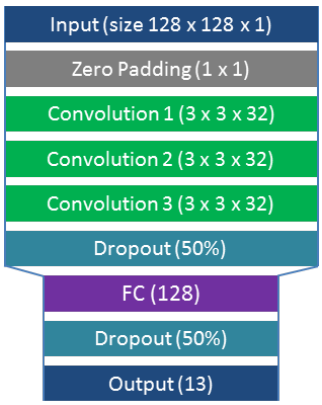

Fig. 6: Architecture of the CNN model.

ReLU non-linearity, and 1 output layer with 13 neurons and Softmax non-linearity. The 13 output neurons are activated correspondingly to the video labels, and an output neuron receives a 1 value if its output is larger than 0.3, which is set empirically. Dropout layers are used to prevent overfitting [25]. An Adamax optimizer [12] is used with a learning rate of 0.002, and a categorical cross entropy is used as a loss function.

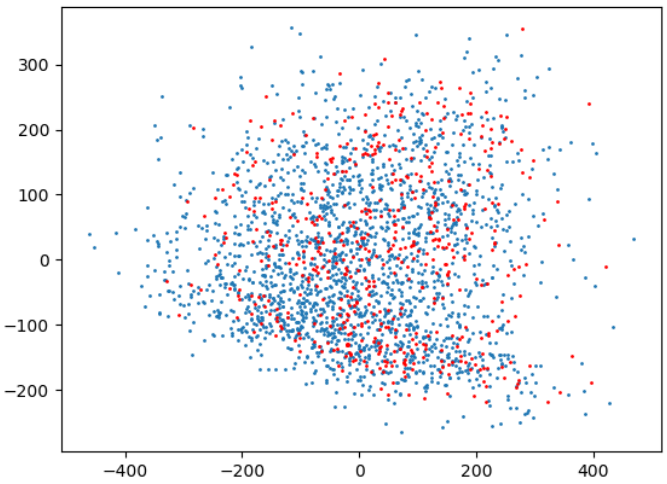

Fig. 7: Projection of training (blue) and testing (red) dataset images into the PCA space.

2) A Principal Component Analysis (PCA) based approach: PCA is an efficient method for dimensionality reduction [9]. It uses the covariance matrix of the data creating a space known as an eigenspace. Each dimension in the space is represented by an eigenvector. The number of eigenvectors required to represent the full data is considerably lower than the dimensionality of the original data. In addition, it is shown that it is possible to interpolate the points created by the projection of the images into a PCA space forming a manifold [20].

In order to apply PCA over our training dataset we combine images into the same array and then compute PCA. Since each image has 128×128 pixels when vectorised becomes 16,384 pixels in a row array, for each image. As a result, we have a $16,384 \times 16,384$ eigenspace matrix, by applying PCA to the covariance matrix.

By projecting the images from the training set into the most significant D_i eigenvectors, we obtain a D_i-dimensional space containing N_{im} images. Each point represents an image. In this work we tested different number of eigenvectors and how it affects the accuracy.

Figure 7 shows 2 dimensions (axes) of these D_1 and D_2, where each point represents one image in our training (blue) and testing (red) dataset.

V. EXPERIMENTAL RESULTS

A. Comparison baseline

We compare our model with a baseline algorithm that uses shallow features [4]. The baseline method works as follows: The video transcript is generated using the CMU Sphinx toolkit [14]. Then, a vector of word frequencies is generated for each transcript. The vector is initially 354,986 dimensional corresponding to a list of most used English words[2], then it

[2]https://github.com/dwyl/english-words

is reduced to a 7,937 dimensional vector by storing only the words that exist more than once in the training dataset's videos. The vector is finally normalized by dividing on the total sum of frequencies.

After extracting word frequencies features vectors, we experiment with different classifiers: multilayer perceptions (MLP), Decision Trees , K-Nearest Neighbors (K-NN), and Random Forests [22].

B. Evaluation metrics

Evaluation of our model against the baseline is done with two metrics: $Label\ Accuracy$ (Eq. 1) expresses the ability of the model to correctly generate a label for a video, so it is penalized every time a single label is incorrect. The $Class\ Accuracy$ (Eq. 2) expresses the ability of the model to correctly generate all the $N = 13$ labels to a video, which means that a classification is considered incorrect as soon as one label is incorrect.

$$Label\ Accuracy = \frac{1}{K}\sum_{k=1}^{K}(1 - \frac{|\overrightarrow{y_{test}} - \overrightarrow{y_{predicted}}|}{N}) \quad (1)$$

$$Class\ Accuracy = \frac{1}{K}\sum_{k=1}^{K} 1, \text{if } \overrightarrow{y_{test}} = \overrightarrow{y_{predicted}} \quad (2)$$

Where K is the number of videos. Both metrics are in $[0, 1]$ and the larger the values the better the performances. Naturally, $Class\ Accuracy$ would be inferior to $Label\ Accuracy$. For instance, if a video is classified as 0100000000000 while its ground truth is 0100000010000, $Label\ Accuracy = \frac{12}{13} = 0.92$ while $Class\ Accuracy = 0$.

C. Results

Figure 8 shows the training progress of our CNN model and the MLP baseline. The performance difference is more noticeable with the $Class\ Accuracy$ metric than with the $Label\ Accuracy$ metric. The proposed model outperforms the baseline using an MLP in terms of $Label\ Accuracy$ slightly and in terms of $Class\ Accuracy$ with more than 9%. After 50 iterations, the baseline shows overfitting and its performance decreases.

Our PCA model classifies the input with the K-Nearest Neighbour (K-NN) algorithm, with $K = 1$, and Euclidean distance [17]. We projected each testing image into the training dataset eigenspace and classified according to the nearest point (shortest Euclidean distance). The recognition accuracy strongly depends on the number of the eigenvectors (dimensions) considered. For example, assuming the number of eigenvectors is $D_i = 15$, we obtain $Class\ Accuracy = 53.16\%$ and $Label\ Accuracy = 93.09\%$. When using more eigenvectors the accuracy increases as well. e.g. for $D_i = 100$ we obtain $Class\ Accuracy = 68.52\%$ and $Label\ Accuracy = 95.43\%$. Figure 9 shows the $Class\ Accuracy$ and $Label\ Accuracy$ according to the number of eigenvectors. As expected, performances improve when more eigenvectors are used.

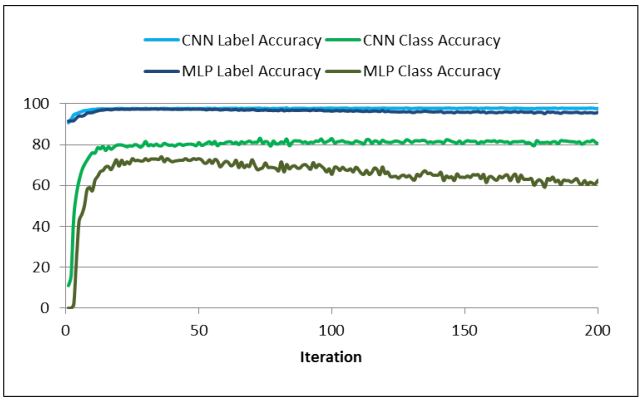

Fig. 8: Metrics values during models training.

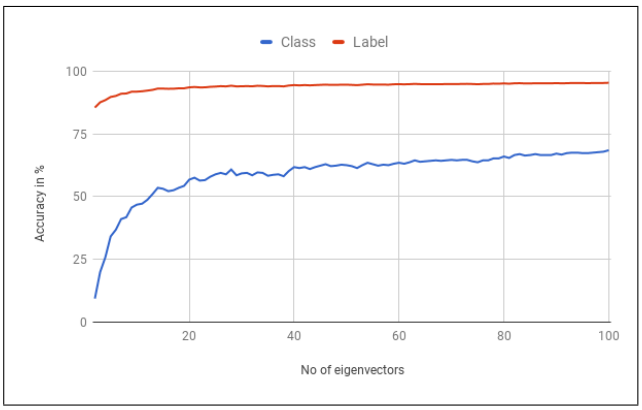

Fig. 9: Accuracy according to the number of eigenvectors.

Table 2 shows best performances of all models that we experimented with. Our CNN model outperforms all the baselines, and the best baseline performances were obtained by an MLP that has 7,937 input neurons, 1 hidden layer with 1024 neurons, and 13 neurons in the output layer. A part from the CNN, PCA and MLP, other baselines using Decision Trees, K-NN and Random Forest have not given satisfactory performances. Trying different configurations of the baseline models, including the MLP (by adding hidden layers and neurons) has not led to improved performances.

It is worth noting that performances of our method are high despite of the imperfection of the speech recognition tool [14] that can cause a word error rate (WER) as high as 26.9% [8]. Given that the KAS dataset is currently single speaker, we explain our model's high performances by the fact that the transcript errors would be associated with certain video topic labels and lead to correct classifications despite of word mistakes.

We also confirmed our hypothesis of using speech transcript instead of using frames. We trained an Alexnet [13] that takes single video frames and produce topic labels, and we prepared a dataset of 1,263,227 frames by extracting equidistant keyframes in 1s intervals from the KAS videos. Frames were resized to 160×112 in order to overcome the high resolution differences (Fig. 4). This classifier gave $Class\ Accuracy$ and

Table 2: Classifier performances using the testing set (Sec. IV). Best ones are the CNN model (after 73 training iterations), the second best the MLP model (after 36 training iterations) and the third PCA (with 100 eigenvectors)

Model	$Label\ Accuracy$ (100%)	$Class\ Accuracy$ (100%)
CNN	97.87%	83.10%
MLP	97.53%	74.08%
PCA	95.43%	68.52%
Decision Trees	90.71%	37.42
K-NN (K=3)	88.51%	22.64%
Random Forest	91.87%	6.33%

$Label\ Accuracy$ values below 0.5, which is way inferior to the models using transcripts.

VI. Conclusion and future work

In this paper, we presented a method for automatic classification of educational videos. Our method works as follows: First, it extracts video transcripts using a standard toolkit. Then, it converts the transcripts into images using a statistical co-occurrence transform. Finally, it uses a supervised learning classifier to generate the video topic labels. In this work, we reported experiments using two types of classifiers, a convolutional neural network (CNN) and a principal component analysis (PCA) model.

We used the Khan Academy on a Stick (KAS) dataset and evaluate our model against a baseline that is based on transcript word frequency feature vectors. Results demonstrate the effectiveness of our approach and a significant increase in performances.

This paper reports one module of our ongoing work to involve content-based indexing in the educational videos in order to enable more intuitive video retrieval. As a future work, we will evaluate our method using datasets collected from different MOOC platforms, and we will continue working towards content-based video retrieval using sketches and audio keywords.

References

[1] S. Abu-El-Haija, N. Kothari, J. Lee, P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016.

[2] O. Adigun and B. Kosko. Using noise to speed up video classification with recurrent backpropagation. In *International Joint Conference on Neural Networks (IJCNN)*, pages 108–115. IEEE, 2017.

[3] S. Basu, Y. Yu, and R. Zimmermann. Fuzzy clustering of lecture videos based on topic modeling. In *International Workshop on Content-Based Multimedia Indexing (CBMI)*, pages 1–6. IEEE, 2016.

[4] D. Brezeale and D. J. Cook. Automatic video classification: A survey of the literature. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(3):416–430, 2008.

[5] J. Chapes. Online video in higher education: Uses and practices. In *EdMedia: World Conference on Educational Media and Technology*, pages 1133–1138. Association for the Advancement of Computing in Education (AACE), 2017.

[6] R. Chaudhry et al. Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1932–1939. IEEE, 2009.

[7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893. IEEE, 2005.

[8] C. Gaida, P. Lange, R. Petrick, P. Proba, A. Malatawy, and D. Suendermann-Oeft. Comparing open-source speech recognition toolkits, 2014.

[9] F. Han and H. Liu. Scale-invariant sparse PCA on high-dimensional meta-elliptical data. *Journal of the American Statistical Association*, 109(505):275–287, 2014.

[10] I. Jargalsaikhan, S. Little, R. Trichet, and N. E. O'Connor. Action recognition in video using a spatial-temporal graph-based feature representation. In *International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE, 2015.

[11] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1725–1732, 2014.

[12] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[14] P. Lamere, P. Kwok, E. Gouvea, B. Raj, R. Singh, W. Walker, M. Warmuth, and P. Wolf. The CMU SPHINX-4 speech recognition system. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 2–5. IEEE, 2003.

[15] I. Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123, 2005.

[16] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE, 2008.

[17] L. Liberti, C. Lavor, N. Maculan, and A. Mucherino. Euclidean distance geometry and applications. *Siam Review*, 56(1):3–69, 2014.

[18] M. Marsden, K. McGuinness, S. Little, and N. E. O'Connor. Holistic features for real-time crowd behaviour anomaly detection. In *International Conference on Image Processing (ICIP)*, pages 918–922. IEEE, 2016.

[19] M. Noer. One man, one computer, 10 million students: How khan academy is reinventing education. www.forbes.com/sites/michaelnoer/2012/11/02/one-man-one-computer-10-million-students-how-khan-academy-is-reinventing-education, Accessed on 19/07/2017.

[20] M. Oliveira, A. Sutherland, and M. Farouk. Manifold interpolation for an efficient hand shape recognition in the irish sign language. In *International Symposium on Visual Computing*, pages 320–329. Springer, 2016.

[21] L. Pappano. The year of the mooc. *The New York Times*, 2(12):2012, 2012.

[22] S. Shalev-Shwartz and S. Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge University Press, 2014.

[23] H. V. Shin, F. Berthouzoz, W. Li, and F. Durand. Visual transcripts: lecture notes from blackboard-style lecture videos. *ACM Transactions on Graphics (TOG)*, 34(6):240, 2015.

[24] K. Soomro, A. R. Zamir, and M. Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[25] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[26] C. Thompson. How khan academy is changing the rules of education. *Wired Magazine*, 126:1–5, 2011.

[27] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu. Action recognition by dense trajectories. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3169–3176. IEEE, 2011.

[28] Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang. Deep networks for image super-resolution with sparse prior. In *International Conference on Computer Vision (ICCV)*, pages 370–378, 2015.

[29] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4694–4702, 2015.