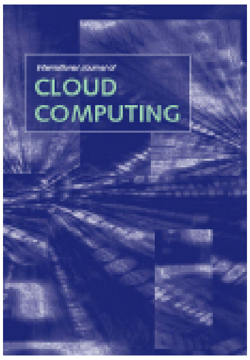


International Journal of Cloud Computing

ISSN online: 2043-9997 - ISSN print: 2043-9989
https://www.inderscience.com/ijcc

KBSS: an efficient approach of extracting text contents from lecture videos - computational intelligence techniques

Sreerama Murthy Velaga, Panigrahi Srikanth, D. Khader Basha

DOI: 10.1504/IJCC.2024.10061837

Article History:

Received:	05 June 2019
Last revised:	16 August 2020
Accepted:	07 January 2021
Published online:	26 January 2024

KBSS: an efficient approach of extracting text contents from lecture videos – computational intelligence techniques

Sreerama Murthy Velaga*

Department of Computer Science and Engineering,
N S Raju Institute of Technology (NSRIT),
Pendurthi, Visakhapatnam-531173, Andhra Pradesh, India
Email: sreeramamurthy.v@gmrit.edu.in
*Corresponding author

Panigrahi Srikanth

Department of Computer Science and Engineering,
Koneru Lakshmaiah Education Foundation,
K L (Deemed to Be University),
Bowrampet, Hyderabad, 500043, India
Email: p.srikanth@klh.edu.in

D. Khader Basha

Department of Computer Science and Engineering,
GMR Institute of Technology,
Rajam 532127, India
Email: khaderdbasha@gmail.com

Abstract: For the last few decades, there is a lot of research going on in the areas of image processing and text mining. They became an emerging research area because an image or a video with cloud is a major source of data, whereas text is a prominent and direct source of information in a video lecture. The challenges usually faced are converting the lecture video frames into binary conversion matrix, extracting image to text matrix, defining the threshold value and classification. Here, in this paper an efficient approach for extracting text contents from metadata lecture videos with cloud is proposed. We built a frame work KBSS in which the frames are converted into binary matrix, then extracted key factors with a text matrix, then clustered with proposed similarity measures in order to reduce the matrix and classification of the text matrix using neural networks, and finally checked the proposed similarity measure with the properties of each case-wise. The objective is to extract text from meta lecture videos with cloud and improving algorithm performance.

Keywords: meta lecture video; computational intelligence techniques; binary matrix; key factors; text and image mining.

Reference to this paper should be made as follows: Velaga, S.M., Srikanth, P. and Basha, D.K. (2024) 'KBSS: an efficient approach of extracting text contents from lecture videos – computational intelligence techniques', *Int. J. Cloud Computing*, Vol. 13, No. 1, pp.1–24.

Biographical notes: Sreerama Murthy Velaga received his PhD in BIT, Mesra, Ranchi in 2011. He has completed his BE in Mepco Schlenk Engineering College, Sivakasi, TN, India in 2000. He is having 20+ years of experience in industry as well as teaching. He is presently working as a Professor at the GMR Institute of Technology, Rajam, AP, India. His area of interest is computer vision, image processing and video surveillance.

Panigrahi Srikanth received his MTech in CSE from the JNTU, Kakinada, AP in 2015 and BTech in IT from the JNTU, Kakinada, AP in 2012. He is having five years of experience in research and teaching. He has a professional membership such as MACM and MIAENG. He is currently working as an Assistant Professor at the GMR Institute of Technology, Rajam, AP, and India. His areas of interest are in computational intelligence, scalable data mining and medical mining.

D. Khader Basha received his BTech in GMR Institute of Technology, Rajam, AP, India in 2020. His area of interest is image processing and business intelligence. He has professional memberships such as member of ACM. He got best project award and best presentation award several universities. His area of interest is computer vision, image processing, video surveillance and business intelligence.

This paper is a revised and expanded version of a paper entitled 'An efficient approach of extracting text contents from lecture videos – computational intelligence techniques' presented at International Conference on Artificial Intelligence and Machine Learning (IAIM2019), Sathyabama Institute of Science and Technology (Deemed to be University), Chennai, India, 3–5 April 2019.

1 Introduction

Nowadays, the usage of modern learning methods, such as e-learning and other is increasing day by day. But, there is a problem that is faced by the explorers during the usage is that lack of information retrieval system. Therefore, there is a lot of work is going on in this research area. This paper aims to design a framework to assist student community or any learner to extract key text information (referred as key factors) from the lecture videos. Initially, the input video is converted into frames, i.e., image, then each frame into binary matrix, then extracting key factors with text matrix. Subsequently, clustering techniques are employed with proposed similarity measures in-order to reduce the size of the matrix and finally classification of text matrix using neural networks.

In detail after converting the frames into binary matrix, pre-processing technique is applied to reduce the size of the matrix. There are many pre-processing techniques like OCR, PCA, SVD, etc. In this framework, we are applying the principal component analysis (PCA) pre-processing technique combined with Hebbian rules.

With the reduced matrix we can obtain the feature vectors and then we can apply classification algorithms. In general, the classification algorithms are back propagation algorithm, etc. In this approach we are using the back propagation neural network (BPN) which is a method to train the neural network in-order to perform the tasks with more accuracy. In order to utilise the computing power of cloud, the proposed methodology is implemented in cloud. The following section highlights importance of using cloud for video and text extraction form video.

1.1 Video and text processing with cloud

1 The performance will be improved with the use of cloud for processing the video and text.

2 Cloud provides a variety of features for video processing like automatic system, security and large network access.

3 The cloud video intelligence processing service provides the feature of extracting the metadata that can be utilised to record your video and make it simple to sort out and look through out your video content.

4 The main advantage of video processing with cloud is that of requires less CPU time for processing.

5 All the video processing tasks are performed with the help of cloud.

6 Video and text processing with cloud uses dimensionality measure to reduce the dimensions.

7 The extraction of frames in the cloud are of high quality and takes less time to extract the frames.

8 The entire process of extracting the text, increasing the accuracy using pre-processing techniques and validating the output is done in cloud.

2 Related work

Balasubramanian et al. (2016) developed a model which is used to extract the set of key phrases and segments that describes the content of a video effectively. An approach called naive Bayes classifier and rule-based classifier is used for the retrieval. Xu and Lee (2015) proposed an approach for dimensionality reduction for regression problems is presented in which clustering is done based on features. A group of clusters are created for a set of training instances and one feature is extracted from each through a weighted combination. Parallelly, the numbers of features are reduced to the number of clusters obtained. Shang et al. (2012) presented an algorithm called graph dual regularisation non-negative matrix factorisation (DNMF) is proposed which considers geometric structures of data and features simultaneously. As an extension to DNMF, DNMF algorithm is presented and in which these would be helpful in scientific computing, data mining and computer vision (Kuang et al., 2018). Here in this paper, a multi-level deep learning algorithm is created in-order to learn deep networks and a tree classifier jointly to support large-scale visual recognition of object classes. A hierarchical structure of concept ontology is constructed to maintain a number of object classes and which automatically determines the inter-related tasks. Gao et al. (2017) proposed an algorithm based on bipartite graphical correlation and implicit trust is developed to address the problem of low prediction accuracy while applying data analysis and behavioural analysis in the present age of big data.

Zhong et al. (2017) developed a semantic, class-specific approach for re-ranking the object proposals is developed which can effectively improve the object detection performance and even consistently can improve the recall performance with fewer proposals. Zhao et al. (2017) developed a deep were hashing method which is able to preserve pyramidal structure as per binary hash codes for retrieval of large-scale image is proposed and in which there is a improvement in retrieval quality because of the use of deep CNNs. Wang et al. (2017) developed a method in which duplicates from collection of documents are detected. This method comprises of feature selection which are nothing but heavy weighted terms, similarity measure and discriminant derivation in which SVM is adopted to learn the discriminant function. Yu et al. (2017) developed an action recognition system is developed based on hybrid network architecture which is a challenging task in computer vision to recognise the actions in a video. This hybrid network consists of two stream CNNs and two stream single layer pi-LSTM and an IDT. Yu et al. (2015) proposed a method to model the image attention and the question attention simultaneously, a network architecture is developed so that the features which are irrelevant are reduced effectively and more discriminative features are extracted. Zhao et al. (2015) proposed a system that helps the construction of fuzzy rule-based system which is done by building sparse LS-SVMs is proposed in this paper. First of all, the primal optimisation problem is presented, and then efficient learning mechanism is then proposed to extract the sparse set.

Zhao et al. (2016) and Guan et al. (2015) proposed a non-negative latent representation learning algorithm for representation learning from multi-view data in which the conceptual latent space of items are tried to learn first and the main theme is to learn a common latent space in different views which captures the semantic relationships in between data items. Other related works with image processing and video extract text contents are Velaga et al. (2009), Velaga et al. (2010a, 2010b) and Velaga et al. (2018a), intelligence techniques (Deverapalli et al., 2016; Deverapalli and Srikanth, 2015; Srikanth and Rajasekhar, 2016), classification techniques (Srikanth and Deverapalli, 2016, 2017; Radhakrishna et al., 2018a, 2018b, 2018c) and clustering techniques (Aljawarneh and Vangipuram, 2020).

3 Methodologies

The methodology or the architecture of the system is as shown in Figure 1. The following steps explain how the proposed methodology happens in AWS cloud:

Step 1 Initialise the Internet gateway via Internet in-order to perform operations using AWS cloud services.

Step 2 Once accessing the AWS cloud services dashboard, create a new instance of EC2 service with a web application.

Step 3 Now create an S3 bucket and EBS volume in-order to store the video files.

Step 4 Now initialise proposed methodology, i.e., KBSS in-order to detect the text contents and extract the text.

Step 5 Once after completing the procedure of KBSS, compute the system performance.

Step 6 Now, suspend the EC2 instance.

Step 7 End of the procedure.

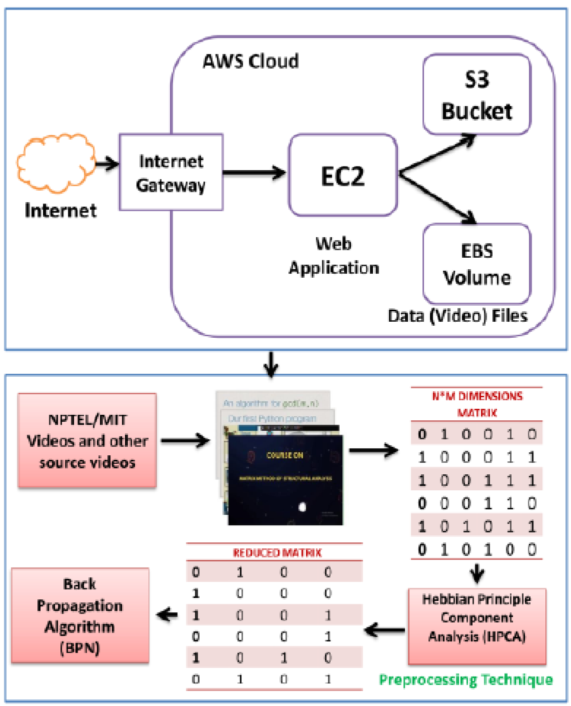
3.1 Input data

As we are constructing a framework to retrieve the information from lecture videos, so we are taking the lecture videos as input which are available in the sources:

1 NPTEL videos available in
- http://digimat.in/ and https://onlinecourses.nptel.ac.in/

2 MIT and Coursera videos available in
- https://ocw.mit.edu/index.htm

Figure 1 Architecture of the system (see online version for colours)



3 Harvard videos available in
- https://courses.my.harvard.edu/psp/courses

4 Oxford videos available in
- https://www.ox.ac.uk/admissions/graduate/courses/introducing-our-courses?wssl=1

5 Someother input videos available in

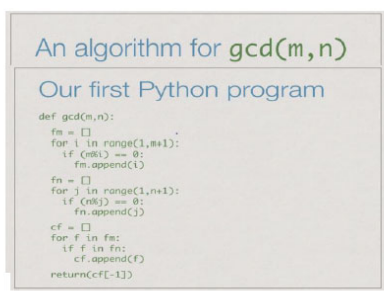
- https://www.ox.ac.uk/admissions/graduate/courses/introducingour-courses?wssl=1

6 Complete all different video courses input recourse link is:
   - https://alison.com/certificate-courses
   - https://www.coursera.org/google
   - http://www.openuniversity.edu/courses
   - https://www.edx.org/
   - https://www.udemy.com/topics/
   - https://onlinecourses.nptel.ac.in/
   - https://www.class-central.com/provider/coursera
   - https://online-learning.harvard.edu/subject/business http://www.indiaeducation.net/onlineeducation/11-free-online-courses-that-you-can-take-up-from-home.htm
   - https://www.upgrad.com/ https://www.who-umc.org/education-training/online-courses/

3.2 Converting into frames

The lecture video which is taken as input is divided or converted into individual frames, i.e., into images because we need to retrieve the information from each slide of lecture video as follows (Figure 2).

Figure 2 Frames obtained from a video (see online version for colours)



3.3 Binary matrix

Previously we have divided the lecture video into frames or images. Now, each frame is converted into binary matrix (0 s and 1 s) of size $n \times m$ (pixel × pixel), where n power p denotes the row pixels and m power p denotes the column pixels. An individual file(.txt) is created for each frame which consists of matrix with binary data (Figure 3).

Figure 3 Binary matrix with n x m dimensions (see online version for colours)

N*M DIMENSIONS MATRIX					
0	1	0	0	1	0
1	0	0	0	1	1
1	0	0	1	1	1
0	0	0	1	1	0
1	0	1	0	1	1
0	1	0	1	0	0

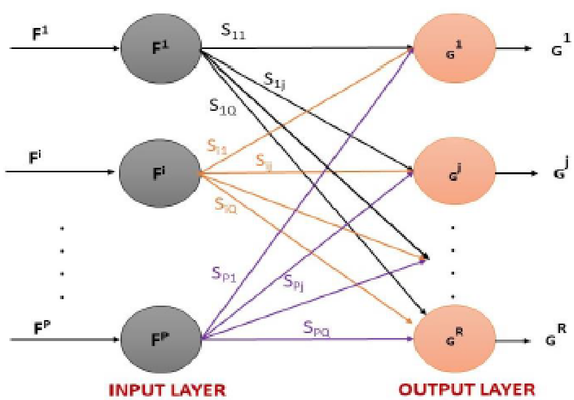
3.4 *Applying pre-processing techniques*

Here we have the input as matrix of size $n \times m$. By applying pre-processing techniques, we can reduce the size of the matrix. There are lot of pre-processing techniques available like OCR, PCA, SVD, etc. In this paper, we are applying HPCA pre-processing technique.

3.4.1 *Hebbian based on PCA architecture*

The architecture of Hebbian network based on PCA is as shown in Figure 4.

Figure 4 Architecture of HPCA (see online version for colours)



3.4.2 *HPCA algorithm*

Step 1 Start the process

Step 2 Initialising the binary matrix.

Step 3 Select the matrix of $P \times R$ dimensions, where P is the number of frames and R contains the data of the frame.

Step 4 Now, apply the neuron model as linear equation, where $i = 1$ to P and $j = 1$ to R.

$$G^j(R) = \sum_{i=1}^{p} S_{ji}(R).F_i(P) \tag{1}$$

where $i = 1$ to P and $j = 1$ to R.

Step 5 Calculating the GHA

$$\Delta s_{ji}(R) = \alpha.G^j(R).\left[F^i(P) - \sum_{j=1}^{R} S_{ji}(R).G^j(R)\right] \tag{2}$$

Step 6 Updating GHA

$$\Delta s_{ji}(R) = \alpha.G^j(R).\left[F^i(P) - \sum_{j=1}^{R} S_{ji}(R).G^j(R)\right] \tag{3}$$

Step 7 Updating GHA one time

$$F^{i\rightarrow}(p) = F^i(p) - \sum_{j=1}^{R} S_{ji}(R) \tag{4}$$

Step 8 Updating GHA two times

$$F^{i\rightarrow\rightarrow}(p) = F^{\rightarrow\rightarrow}(p) - \sum_{j=1}^{R} S_{ji}(R) \tag{5}$$

Step 9 Updating GHA two times with inverse.

$$F_j'^{\rightarrow}(p) = F_j^{\rightarrow}(p) - \sum_{j=1}^{R} S_j^{\rightarrow}.G_R(j) \tag{6}$$

Step 10 Consider two cases.

- *Case 1:* observing at neurons $j = 1$ and updating weights. It is considered as principle component.

$$F_j'^{\rightarrow}(p) = F_j^{\rightarrow}(p) - \sum_{j=1}^{R} S_j^{\rightarrow}.G_R(1) \tag{7}$$

- *Case 2:* observing at neurons $j = 2$ and updating weights. It is considered as principle component.

$$F_j'^{\rightarrow}(p) = F_j^{\rightarrow}(p) - \sum_{j=1}^{R} S_j^{\rightarrow}.G_R(2) \tag{8}$$

Following this case based on finding eigen vector and correlation matrix.

Step 11 Reducing the learning rate and stopping the process.

$$\alpha.(t+1) = 0.5\alpha.(t). \tag{9}$$

3.5 Reduced matrix

After applying the HPCA pre-processing technique, the original matrix generated from each frame of size $n \times m$ is reduced into $n \times j$ dimensional matrix where $j < m$, where n power p denotes the row pixels and j power p denotes the column pixels as shown in Figure 5.

Figure 5 Reduced matrix (see online version for colours)

REDUCED MATRIX			
0	1	0	0
1	0	0	0
1	0	0	1
0	0	0	1
1	0	1	0
0	1	0	1

3.6 Applying back propagation algorithm

Taking the reduced matrix as input we are applying back propagation algorithm. Now, let us see the flow chart, architecture and algorithm of back propagation.

3.6.1 Flow chart of back propagation algorithm

Flow chart of back propagation algorithm is shown in Figure 6.

Figure 6 Flow chart of BPN algorithm (see online version for colours)

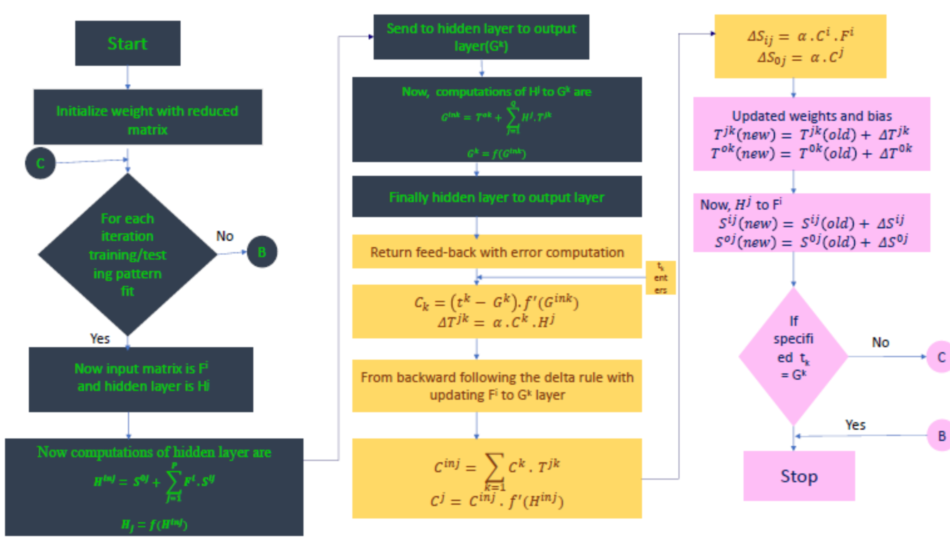
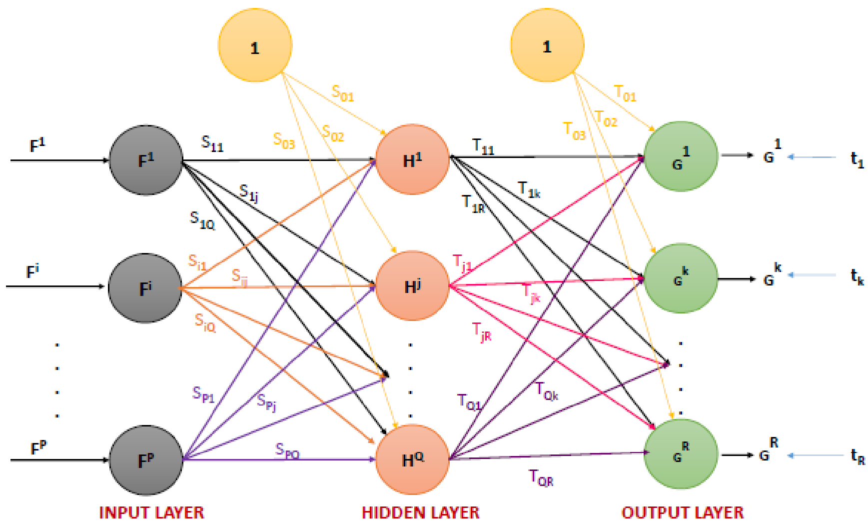


Figure 7 Architecture of BPN algorithm (see online version for colours)



3.6.2 Back propagation architecture

Figure 7 shown as back propagation architecture.

3.6.3 Back propagation algorithm

Step 1 Start the process.

Step 2 Initialising the weights with reduced matrix.

Step 3 If there is a training or testing pattern fir for each iteration, then take the input matrix as F_i and transfer it to the hidden layer as H_j. Else stop.

Step 4 Now, computations of hidden layer are done. They are:

$$H^{inj} = S^{0j} + \sum_{j=1}^{p} F^{i}.S^{ij} \tag{10}$$

and

$$H_j = f\left(H^{inj}\right), \text{ where } i = 1 \text{ to } P and j = 1 \text{ to } Q. \tag{11}$$

Step 5 Send the output computation results H_j which is done in above step to the output layer G_K.

Step 6 Now, calculate the output signal from output layer.

They are

$$G^{inj} = T^{0k} + \sum_{j=1}^{Q} H^{j}.T^{jk} \tag{12}$$

and

$$G_K = f\left(G^{ink}\right), k = 1 \text{ to } R \tag{13}$$

Step 7 The output is compared with the target pairs and the error correction factor is calculated, the error correction factor result is propagated back into the network. The error correction factor between output and hidden layers is calculated using:

$$C_k = \left(t^k - G^k\right).f'\left(G^{ink}\right) \tag{14}$$

Step 8 Now, calculate the weight and bias correction factors using:

$$\Delta T^{jk} = \alpha.C^k.H^j. \tag{15}$$

and

$$\Delta T^{0k} = \alpha.C^k. \tag{16}$$

Step 9 Following the delta rule, compute the error correction factor between hidden and output layer using:

$$C^{inj} = \sum_{k=1} C^k.T^{jk}, \Delta S^{ij} = \alpha.C^i.F^i \tag{17}$$

and

$$C^j = C^{inj}.f'\left(H^{inj}\right) \tag{18}$$

Step 10 Now, compute the change in weights and bias based on:

$$\Delta S^{ij} = \alpha.C^i.F^i \tag{19}$$

and

$$\Delta S^{0j} = \alpha.C^i \tag{20}$$

Step 11 Update weight and bias on output unit as following

$$T^{jk}(new) = T^{jk}(old) + \Delta T^{jk} \tag{21}$$

and

$$T^{0k}(new) = T^{0k}(old) + \Delta T^{0k} \tag{22}$$

Step 12 Update weight and bias on output unit as following:

$$S^{ij}(new) = S^{ij}(old) + \Delta S^{ij} \tag{23}$$

and

$$S^{0j}(new) = S^{0j}(old) + \Delta S^{0j} \tag{24}$$

Step 13 if

$$t_k = G_k \tag{25}$$

Then stop the process, else go to step 3.

4 Case study

4.1 Case 1

Consisting of input frame from sample video 1 when applied with different techniques as represents in detail as Figure 8.

Time stamp 05:06

Dimensions 854×480.

4.2 Case 2

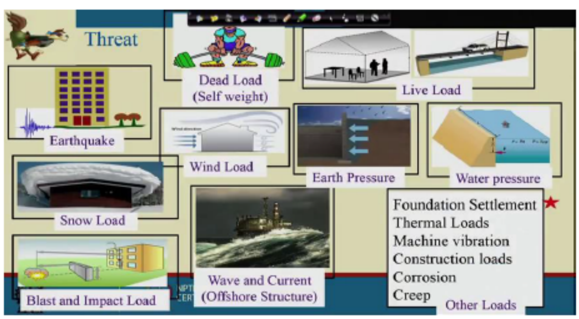
Consisting of input frame from sample video 2 when applied with different techniques as represents in detail as Figure 9.

Time stamp 00:12

Dimensions 640×360.

Figure 8 Consisting of input frame from sample video 1 when applied with different techniques (see online version for colours)

Case 1: Video 1
Input Frame:



Input frame from sample video 1

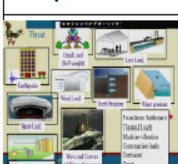
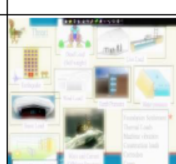
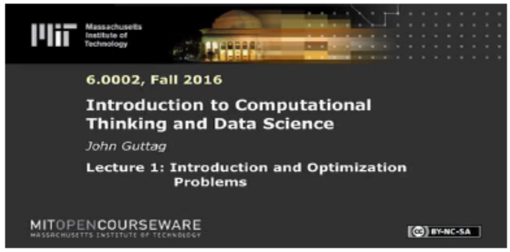
VIDEO-1 (SOURSE FILE: https://onlinecourses.nptel.ac.in/)			
Input Frames	**Without HPCA**	**With HPCA**	**BPN**
			

Figure 9 Consisting of input frame from sample video 2 when applied with different techniques (see online version for colours)

Input Frame:



Input frame from sample video 2

VIDEO- 2 (SOURSE FILE: https://ocw.mit.edu/index.htm)			
Input Frames	Without HPCA	With HPCA	BPN
			

4.3 Case 3

Consisting of input frame from sample video 3 when applied with different techniques as represents in detail as Figure 10.

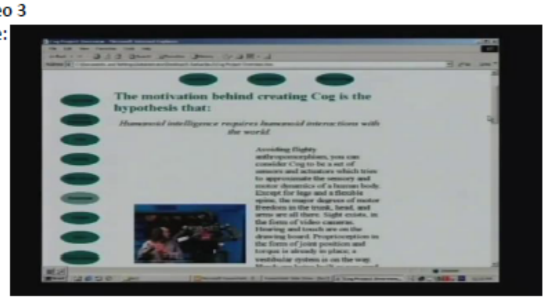
Time stamp 15:313

Dimensions 640×480.

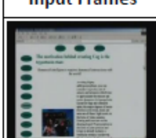
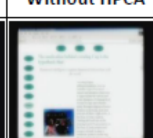
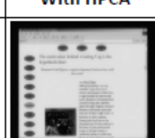
Figure 10 Consisting of input frame from sample video 3 when applied with different techniques (see online version for colours)

Case 3: Video 3

Input Frame:



Input frame from sample video 3

VIDEO- 3 (SOURSE FILE: https://onlinecourses.nptel.ac.in/)			
Input Frames	Without HPCA	With HPCA	BPN
			

4.4 Case 4

Consisting of input frame from sample video 4 when applied with different techniques as represents in detail as Figure 11.

Time stamp 40:08

Dimensions 640 × 480.

4.5 Case 5

Consisting of input frame from sample video 5 when applied with different techniques as represents in detail as Figure 12.

Time stamp 12:33

Dimensions 640 × 360.

Figure 11 Consisting of input frame from sample video 4 when applied with different techniques (see online version for colours)

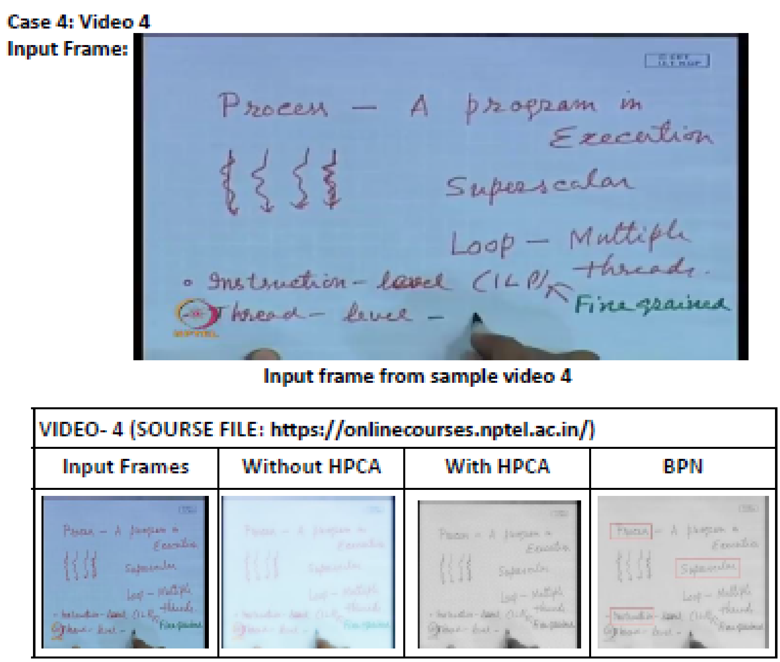
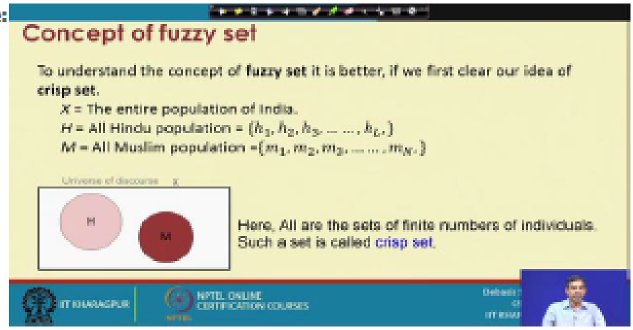


Figure 12 Consisting of input frame from sample video 5 when applied with different techniques (see online version for colours)

Case 5: Video 5

Input Frame:



Input frame from sample video 5

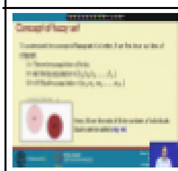
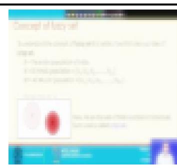
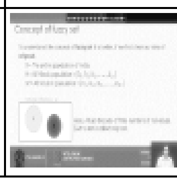
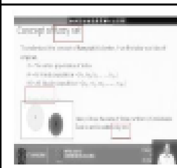
VIDEO- 5 (SOURSE FILE: https://onlinecourses.nptel.ac.in/)			
Input Frames	**Without HPCA**	**With HPCA**	**BPN**
			

Figure 13 Video source 1 with extract text contents using KBSS (see online version for colours)

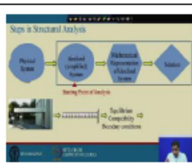
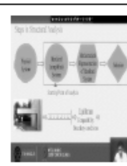
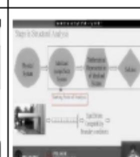
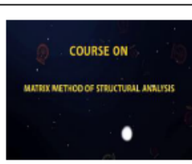
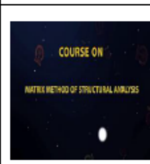
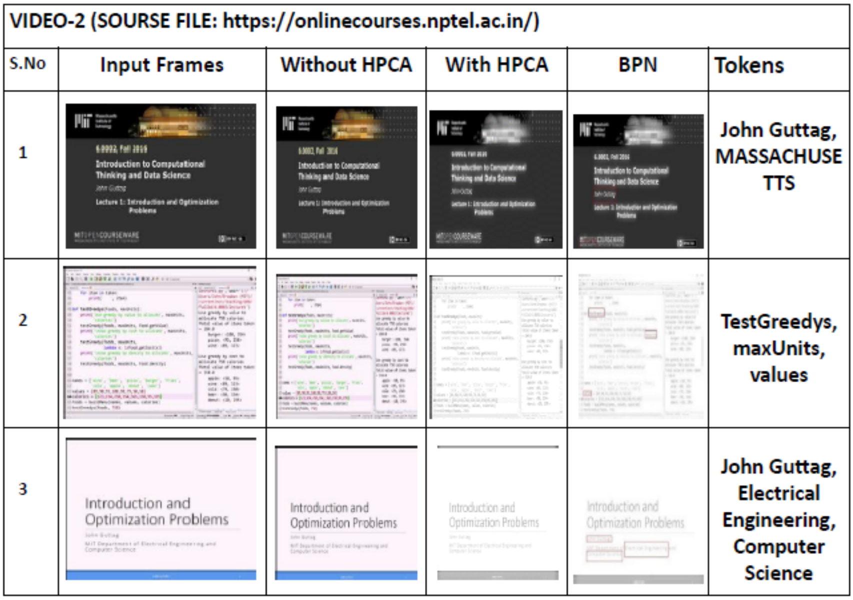
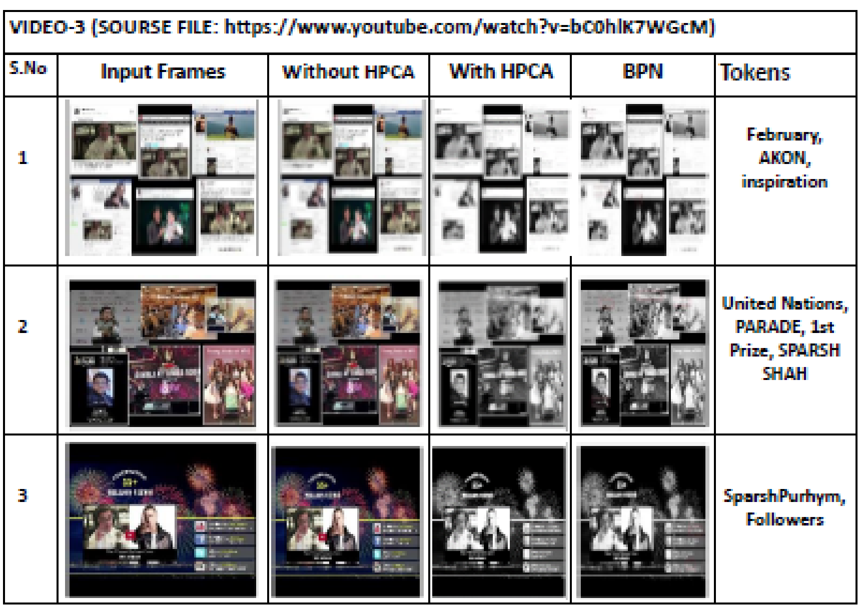
VIDEO-1 (SOURSE FILE: https://onlinecourses.nptel.ac.in/)					
S.No	**Input Frames**	**Without HPCA**	**With HPCA**	**BPN**	**Tokens**
1					Blast and Impact Load, Other Loads
2					Starting Point of Analysis, NPTEL
3					COURSE ON, ANALYSIS

Figure 14 Video source 2 with extract text contents using KBSS (see online version for colours)



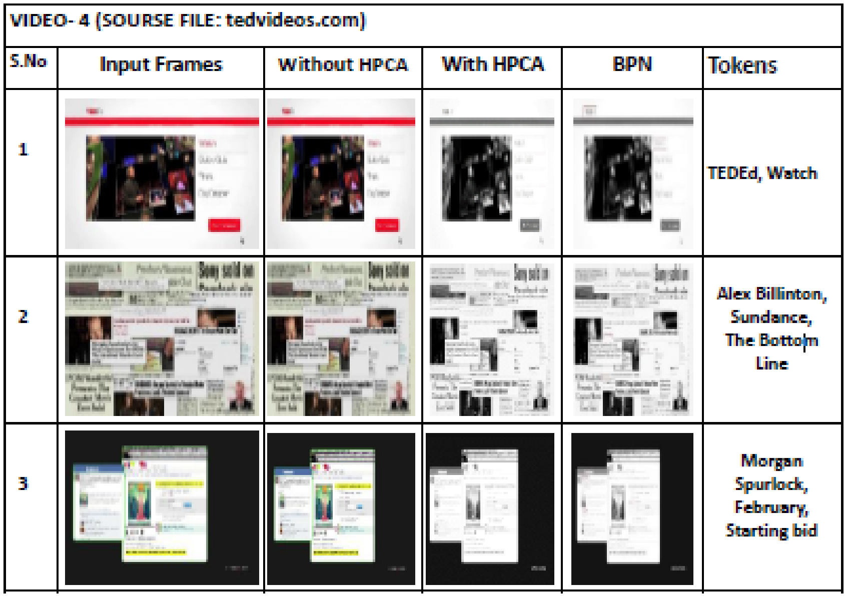
VIDEO-2 (SOURSE FILE: https://onlinecourses.nptel.ac.in/)					
S.No	Input Frames	Without HPCA	With HPCA	BPN	Tokens
1					John Guttag, MASSACHUSETTS
2					TestGreedys, maxUnits, values
3					John Guttag, Electrical Engineering, Computer Science

Figure 15 Video source 3 with extract text contents using KBSS (see online version for colours)



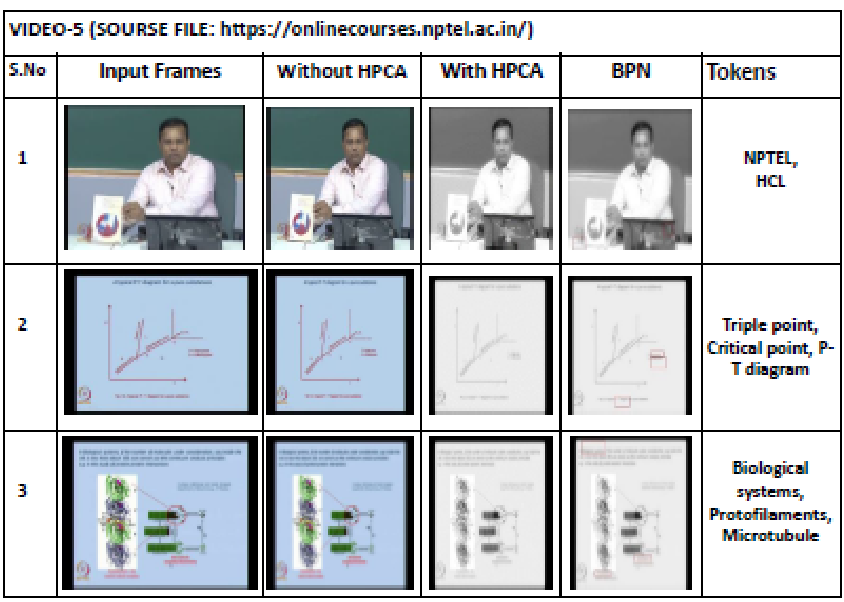
VIDEO-3 (SOURSE FILE: https://www.youtube.com/watch?v=bC0hlK7WGcM)					
S.No	Input Frames	Without HPCA	With HPCA	BPN	Tokens
1					February, AKON, inspiration
2					United Nations, PARADE, 1st Prize, SPARSH SHAH
3					SparshPurhym, Followers

Figure 16 Video source 4 with extract text contents using KBSS (see online version for colours)



VIDEO- 4 (SOURSE FILE: tedvideos.com)					
S.No	Input Frames	Without HPCA	With HPCA	BPN	Tokens
1					TEDEd, Watch
2					Alex Billinton, Sundance, The Bottom Line
3					Morgan Spurlock, February, Starting bid

Figure 17 Video source 5 with extract text contents using KBSS (see online version for colours)



VIDEO-5 (SOURSE FILE: https://onlinecourses.nptel.ac.in/)					
S.No	Input Frames	Without HPCA	With HPCA	BPN	Tokens
1					NPTEL, HCL
2					Triple point, Critical point, P-T diagram
3					Biological systems, Protofilaments, Microtubule

Figure 18 Video source 6 with extract text contents using KBSS (see online version for colours)

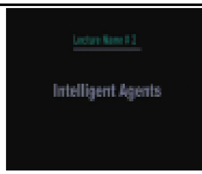
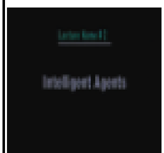
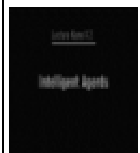
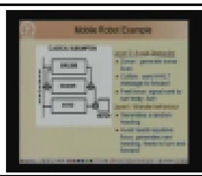
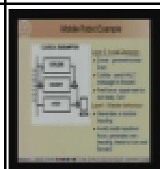
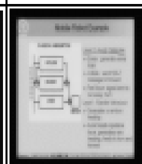
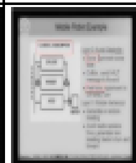
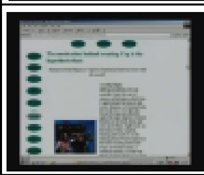
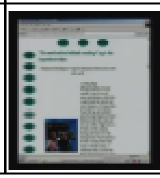
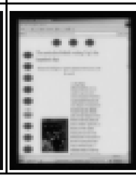
VIDEO- 6 (SOURSE FILE: https://onlinecourses.nptel.ac.in/)					
S.No	Input Frames	Without HPCA	With HPCA	BPN	Tokens
1					Lecture Name, Agents
2					CLASSICAL SUBSUMPTION, Sonar, Feel force
3					Motivation, Avoiding flighty

Figure 19 Video source 7 with extract text contents using KBSS (see online version for colours)

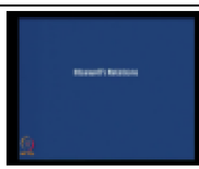
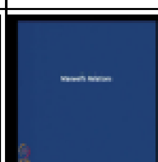
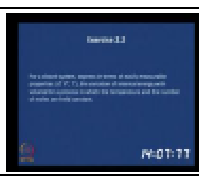
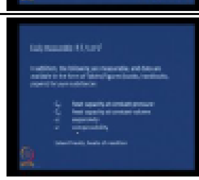
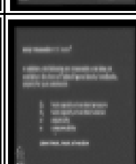
VIDEO-7 (SOURSE FILE: https://onlinecourses.nptel.ac.in/)					
S.No	Input Frames	Without HPCA	With HPCA	BPN	Tokens
1					Maxwell's Relation
2					Exercise 2.2, 14:07:77
3					Pressure, volume

Figure 20 Video source 8 with extract text contents using KBSS (see online version for colours)

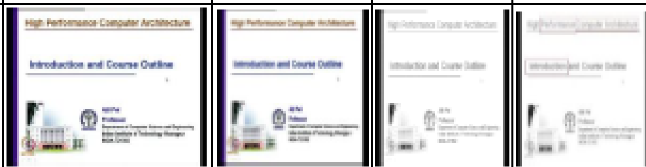
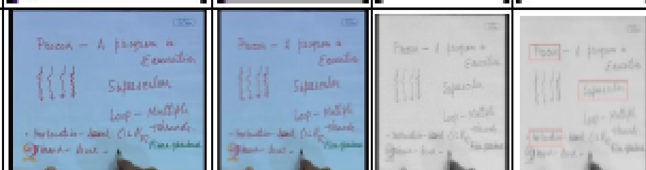
VIDEO- 8 (SOURSE FILE: https://onlinecourses.nptel.ac.in/)					
S.No	Input Frames	Without HPCA	With HPCA	BPN	Tokens
1					Performance, Computer Architecture, Introduction
2					Process, Superscalar, Instruction
3					MOORE'S LAW, Dual-Core, transistors

Figure 21 Video source 9 with extract text contents using KBSS (see online version for colours)

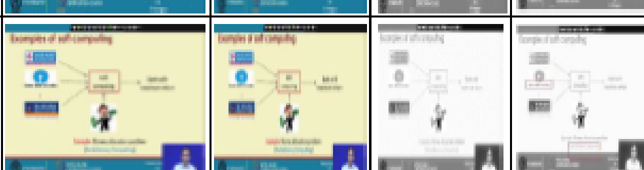
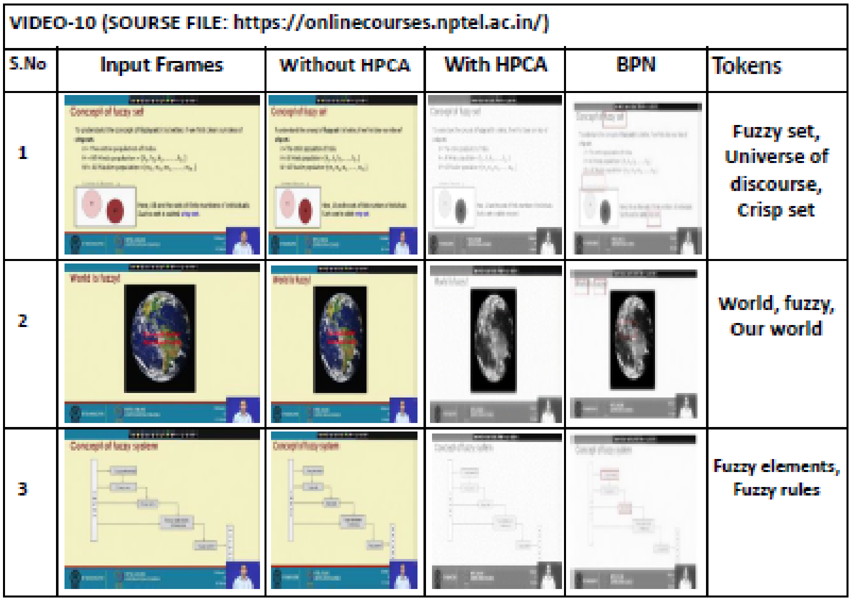
VIDEO-9 (SOURSE FILE: https://onlinecourses.nptel.ac.in/)					
S.No	Input Frames	Without HPCA	With HPCA	BPN	Tokens
1					Introduction, Soft Computing
2					Hybrid computing, HC, SC
3					State Bank of India, Evolutionary Computing

Figure 22 Video source 10 with extract text contents using KBSS (see online version for colours)



5 Results and discussion

In-order to extract the text content from metadata lecture videos, the framework KBSS is proposed in which the frames are converted into binary matrix, then extracting the key factors with text matrix, then applying clustering with proposed similarity measures in order to reduce the matrix and finally classifying the text matrix using neural networks. The results obtained after performing the proposed methodology (framework) are shown in Figures 13–22.

Figure 23 Normalisation with min max and BPN accuracies (see online version for colours)

	Video 1	Video 2	Video 3	Video 4	Video 5	Video 6	Video 7	Video 8	Video 9	Video 10
Numbers of nodes in Input layer	10	10	10	10	10	10	10	10	10	10
Numbers of nodes in Hidden layer 01	8	8	8	8	8	8	8	8	8	8
Numbers of nodes in Hidden layer 02	5	5	5	5	5	5	5	5	5	5
Numbers of nodes in Output Layer	1	1	1	1	1	1	1	1	1	1
Activation function	Sigmoid with Binary	Sigmoid with Binary	Sigmoid with Binary	Sigmoid with Binary	Sigmoid with Binary	Sigmoid with Binary	Sigmoid with Binary	Sigmoid with Binary	Sigmoid with Binary	Sigmoid with Binary
Normalization of Data (N)	Min-Max	Min-Max	Min-Max	Min-Max	Min-Max	Min-Max	Min-Max	Min-Max	Min-Max	Min-Max
Learning rate	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7
Momentum term	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2
Accuracy - Sigmoid with Binary	93.5	92	92.7	93.5	93.96	93.65	94	93.5	92.85	91.99
Accuracy - Sigmoid with Bipolar	94	92.5	93	93.85	94	94	95.12	94.3	93.2	92.5

Figure 24 Proposed approach KBSS accuracies (see online version for colours)

Proposed Work	Video 1	Video 2	Video 3	Video 4	Video 5	Video 6	Video 7	Video 8	Video 9	Video 10
Numbers of nodes in Input layer	10	10	10	10	10	10	10	10	10	10
Numbers of nodes in Hidden layer 01	8	8	8	8	8	8	8	8	8	8
Numbers of nodes in Hidden layer 02	5	5	5	5	5	5	5	5	5	5
Numbers of nodes in Output Layer	1	1	1	1	1	1	1	1	1	1
Activation function	Sigmiod with Binary	Sigmiod with Binary	Sigmiod with Binary	Sigmiod with Binary	Sigmiod with Binary	Sigmiod with Binary	Sigmiod with Binary	Sigmiod with Binary	Sigmiod with Binary	Sigmiod with Binary
Preprocessing	HPCA	HPCA	HPCA	HPCA	HPCA	HPCA	HPCA	HPCA	HPCA	HPCA
Learning rate	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7
Momentum term	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2
HPCA with Accuracy - Binary	94	95	94.98	94.82	95.2	94.66	94.25	94.5	93.22	93
HPCA with Accuracy - Bipolar	94.2	95.6	95.3	95	95	95.2	96	95.2	94.85	94

5.1 Experimental results 1

The proposed methodology KBSS is applied on different videos from different video sources. In particular this methodology is applied on 10 videos which are of both high, low quality and the text from those videos have been extracted. As mentioned in methodology that we have used back propagation algorithm, this BPN algorithm consists of ten nodes in input layer, eight nodes in first hidden layer, five nodes in second hidden layer and one node in output layer. The activation functions we have used are Sigmoid with binary and bipolar with learning rate 0.5 to 0.7, moment factor 0.2, normalisation with min-max and HPCA. In-particular they are shown as figures from Figures 23–25.

Figure 25 Comparison accuracies normalisation and KBSS (see online version for colours)

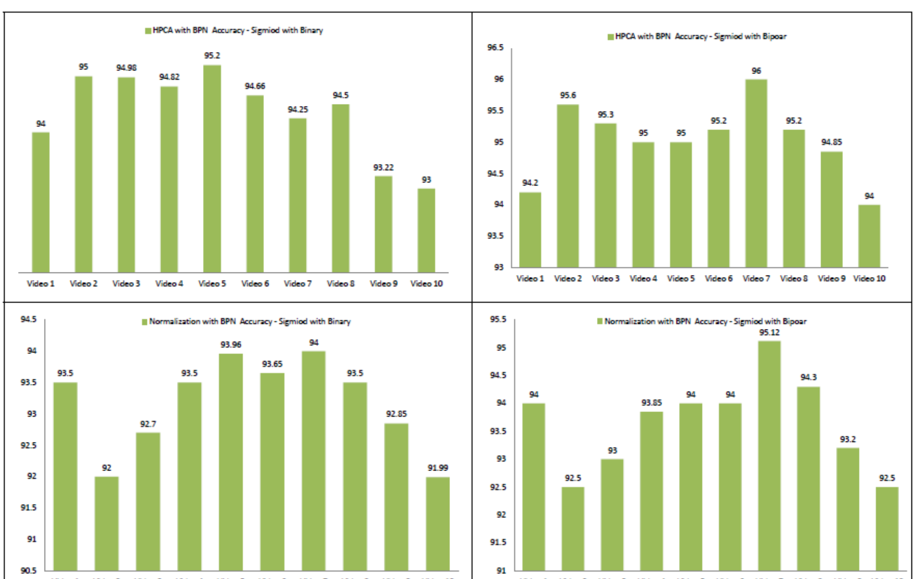


Table 1 Comparison existing and proposed system with text detection and extraction

Author	*Input (videos)*	*Techniques*	*Output*	*Accuracies*
Shivakumara et al. (2010)	Own dataset	Proposed (heuristic rules (filters + edge))	Scene text and graphic text	0.862
Shivakumara et al. (2010)	ICPR 2000	Proposed (heuristic rules (filters + edge))	Scene text and graphic text	0.856
Phan et al. (2014)	TRECVID 2005 and 2006	Semiautomatic system	Text detection and recognition	0.82
Tian et al. (2018)	ICDAR 2011 – USTB-VidTEXT	Generic Bayesian – novel	Text detection and recognition	0.93
Raghunandan et al. (2019)	ICDAR (2013)	INNS+MNNP	Text detection and recognition	0.845
Raghunandan et al. (2019)	ICDAR (2015)	INNS + MNNP	Text detection and recognition	0.665
Raghunandan et al. (2019)	YVT	INNS + MNNP	Text detection and recognition	0.81
Raghunandan et al. (2019)	SVT	INNS + MNNP	Text detection and recognition	0.61
Raghunandan et al. (2019)	MSRA	INNS + MNNP	Text detection and recognition	0.724
Roy et al. (2018)	Youtube + internet sources	RFSC	Text detection and recognition	0.88
Proposed work	Youtube + internet sources + NPTEL	KBSS (proposed)	Text detection and extraction	0.945

6 Discussion

The proposed methodology KBSS which can be applied to extract the text contents from metadata lecture videos, has been experimented on ten different videos from different sources and these videos includes both high quality and low quality. We have applied two procedures namely, the first procedure is Normalisation with min max along with BPN (activation functions – sigmoid with binary, bipolar) and the second procedure is our proposed methodology, i.e., KBSS (sigmoid with binary and bipolar).

After applying the procedures on ten videos, the best accuracies we have got based on each procedure is discussed over here. With the procedure normalisation with min max, BPN with sigmoid binary as activation function the best accuracy we have got is 94 (by applying procedure on sample video 7). With the procedure normalisation with min max, BPN with sigmoid bipolar as activation function the best accuracy we have got is 95.12 (by applying procedure on sample video 7). With the procedure KBSS, i.e., HPCA, BPN with sigmoid binary as activation function the best accuracy we have got is 95.2 (by applying procedure on sample video 5). With the procedure KBSS, i.e., HPCA, BPN with sigmoid bipolar as activation function the best accuracy we have got is 96 (by applying procedure on sample video 7).

7 Conclusions

The current research work is going on the areas of image processing and text mining with cloud. Image or a video is a major source of data, whereas text is a prominent and direct source of information in a video lecture. The challenges that usually faced are converting the lecture video frames into binary conversion matrix, extracting image to text matrix, defining the threshold value and classification. We can design the framework KBSS in which the frames are converted into binary matrix, then extracting key factors with text matrix, then applied clustering with proposed similarity measures in-order to reduce the matrix and classification of text matrix using neural networks and checking proposed similarity measure with properties of each case wise so that the text is extracted from meta lecture videos and improving performance analysis with accuracy and error rate.

Acknowledgements

This work is primarily supported by Department of Science and Technology, Science for Equity Empowerment and Development Division, Government of India with the wide Number SEED/TIDE/158/2016.

References

Aljawarneh, S.A. and Vangipuram, R. (2020) 'GARUDA: Gaussian dissimilarity measure for feature representation and anomaly detection in internet of things', *The Journal of Supercomputing*, Vol. 76, No. 6, pp.4376–4413.

Balasubramanian, V., Doraisamy, S.G. and Kanakarajan, N.K. (2016) 'A multimodal approach for extracting content descriptive metadata from lecture videos', *Journal of Intelligent Information System*, Vol. 46, No. 1, pp.121–145.

Deverapalli, D., Srikanth, P. and Rao, M.R.N. (2016) 'Identification of AIDS disease severity based on computational intelligence techniques using clonal selection algorithm', *International Journal of Convergence Computing*, Vol. 2, Nos. 3–4, pp.193–207, Inderscience Publications.

Deverapalli, D. and Srikanth, P. (2015) 'Identification of aids disease using genetic algorithm', *International conferences on Computational Intelligence and Soft Computing – (IBCB 2015)*, Springer, pp.99–111.

Gao, Q., Gao, L., Fan, J. and Ren, J. (2017) 'A preference elicitation method based on bipartite graphical correlation and implicit trust', *Neurocomputing*, Vol. 237, No. 6, pp.92–100.

Guan, Z., Zhang, L., Peng, J. and Fan, J. (2015) 'Multi-view concept learning for data representation', *IEEE Transactions on Knowledge and Data Engineering*, November, Vol. 27, No. 11, pp.3016–3028.

Kuang, Z., Yu, J., Li, Z., Zhang, B. and Fand, J. (2018) 'Integrating multi-level deep learning and concept ontology for large-scale visual recognition', *Pattern Recognition*, Vol. 78, No. 1, pp.198–214.

Phan, T.Q., Shivakumara, P., Bhowmick, S., Li, S. and Tan, C.L. (2014) 'Semiautomatic ground truth generation for text detection and recognition in video images', *IEEE Transactions on Circuits and Systems for Video Technology*, August, Vol. 24, No. 8, pp.1277–1287.

Radhakrishna, V., Aljawarneh, S.A., Kumar, P.V. and Choo, K-K.R. (2018a) 'A novel fuzzy Gaussian-based dissimilarity measure for discovering similarity temporal association patterns', *Journal of Soft Computing*, Vol. 22, No. 6, pp.1903–1919.

Radhakrishna, V., Aljawarneh, S.A., Kumar, P.V. and Janaki, V. (2018b) 'A novel fuzzy similarity measure and prevalence estimation approach for similarity profiled temporal association pattern mining', *Future Generation Computer Systems*, Vol. 83, pp.582–595.

Radhakrishna, V., Kumar, P.V. and Janaki, V. (2018c) 'SRIHASS – a similarity measure for discovery of hidden time profiled temporal associations', *Multimedia Tools and Applications*, July, Vol. 77, No. 14, pp.17643–17692.

Raghunandan, K.S., Shivakumara, P., Roy, S., Kumar, G.H. and Pal, U. (2019) 'Multi-script oriented text detection and recognition in video/scene/born digital images', *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 29, No. 4, pp.1145–1162.

Roy, S., Shivakumara, P., Jain, N., Khare, V., Dutta, A., Pal, U. and Lu, T. (2018) 'Rough-fuzzy based scene categorization for text detection and recognition in video', *Pattern Recognition*, Vol. 80, No. 8, pp.64–82.

Shang, F., Jiao, L.C. and Wang, F. (2012) 'Graph dual regularization non-negative matrix factorization for co-clustering', *Pattern Recognition*, Vol. 45, No. 6, pp.2237–2250.

Shivakumara, P., Huang, W., Phan, T.Q. and Tan, C.L. (2010) 'Accurate video text detection through classification of low and high contrast images', *Pattern Recognition* Vol. 43, No. 6, pp.2165–2185.

Srikanth, P. and Devrapalli, D. (2016) 'A critical study of classification algorithms using diabetes diagnosis', *2016 IEEE 6th International conferences on Advanced Computing (IACC 2016)*, IEEE 2016, (Google Scholar), 27–28 February, pp.245–249.

Srikanth, P. and Devrapalli, D. (2017) 'CFTDISM: clustering financial text documents using improved similarity measure, *2017 IEEE International Conferences on Computational Intelligence and Computing Research (ICCIC 2017)*.

Srikanth, P. and Rajasekhar, N. (2016) 'A novel cluster evolution for gene miRNA interactions documents using improved similarity measure', *International Conferences on Engineering and MIC-2016 (ICEMIS-2016)*, IEEE-Morocco Section, pp.1–7.

Tian, S., Yin, X-C., Su, Y. and Hao, H-W. (2018) 'A unified framework for tracking based text detection and recognition from web videos', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 40, No. 3, pp.542–554.

Velaga, S.M., Gupta, S. and Mohanta, D.K. (2009) 'Distribution system insulator monitoring using video surveillance and support vector machines for complex background images', *International Journal of Power and Energy Conversion*, Vol. 1, No. 1, pp.49–72.

Wang, Z., Du, C., Fan, J. and Xing, Y. (2017) 'Ranking influential nodes in social networks based on node position and neighborhood', *Neurocomputing*, Vol. 260, No. 10, pp.466–477.

Xu, R-F. and Lee, S-J. (2015) 'Dimensionality reduction by feature clustering for regression problems', *Information Sciences*, Vol. 299, No. 4, pp.42–57.

Yu, S., Cheng, Y., Xie, L., Luo, Z., Huang, M. and Li, S. (2017) 'A novel recurrent hybrid network for feature fusion in action recognition', *Journal of Visual Communication and Image Representation*, Vol. 49, No. 4, pp.192–203.

Yu, Z., Yu, J., Xiang, C., Fan, J. and Tao, D. (2015) 'Beyond bilinear: generalized multi-modal factorized high-order pooling for visual question answering', *Journal of Latex Class Files*, August, Vol. 14, No. 12.

Zhao, W., Luo, H., Peng, J. and Fan, J. (2017) 'Spatial pyramid deep hashing for large-scale image retrieval', *Neurocomputing*, Vol. 243, No. 6, pp.166–173.

Zhao, W., Meng, Q. and Chung, P.W.H. (2016) 'A heuristic distributed task allocation method for multivehicle multitask problems and its application to search and rescue scenario', *IEEE Transactions on Cybernetics*, April, Vol. 46, No. 4, pp.902–915.

Zhao, W., Zhang, J. and Li, K. (2015) 'An efficient LS-SVM-based method for fuzzy system construction', *IEEE Transactions on Fuzzy Systems*, June, Vol. 23, No. 3, pp.627–643.

Zhong, Z., Lei, M., Cao, D., Fan, J. and Li, S. (2017) 'Class-specific object proposals re-ranking for object detection in automatic driving', *Neurocomputing*, Vol. 242, No. 6, pp.187–194.