# Learning Compact Spatio-Temporal Features for Fast Content based Video Retrieval

**Vidit Kumar, Vikas Tripathi, Bhaskar Pant**

***Abstract: Videos are recorded and uploaded daily to the sites like YouTube, Facebook etc. from devices such as mobile phones and digital cameras with less or without metadata (semantic tags) associated with it. This makes extremely difficult to retrieve similar videos based on this metadata without using content based semantic search. Content based video retrieval is problem of retrieving most similar videos to a given query video and has wide range of applications such as video browsing, content filtering, video indexing, etc. Traditional video level features based on key frame level hand engineered features which does not exploit rich dynamics present in the video. In this paper we propose a fast content based video retrieval framework using compact spatio-temporal features learned by deep learning. Specifically, deep CNN along with LSTM is deploy to learn spatio-temporal representations of video. For fast retrieval, binary code is generated by hashing learning component in the framework. For fast and effective learning of hash code proposed framework is trained in two stages. First stage learns the video dynamics and in second stage compact code is learn using learned video's temporal variation from the first stage. UCF101 dataset is used to test the proposed method and results compared by other hashing methods. Results show that our approach is able to improve the performance over existing methods.***

***Keywords: CNN, LSTM, Hashing, CBVR, Deep learning.***

## I. INTRODUCTION

Video has become an important element of multimedia computing and communication environments due to cheap devices like digital cameras, smart phones, etc. Due to these advances in transmission technologies, we are seeing the abrupt growth of videos in the social networking sites with less or without semantic tags associated with it. According to YouTube statistics, every minute about 200 hours of video content is being uploaded to a website like YouTube and similarly around 11 million videos are posted daily in twitter without texts or with poor semantic tags.

**Revised Manuscript Received on December 30, 2019.**
***** Correspondence Author

**Vidit Kumar***, Department of Computer Science and Engineering, Graphic Era deemed to be university, Dehradun, India. Email: viditkumaruit@gmail.com

**Vikas Tripathi,** Department of Computer Science and Engineering, Graphic Era deemed to be university, Dehradun, India. Email: vikastripathi.be@gmail.com

**Bhaskar Pant,** Department of Computer Science and Engineering, Graphic Era deemed to be university, Dehradun, India. Email: pantbhaskar2@gmail.com

Because of this explosive growth of online videos without semantic tags there is need of content based video retrieval (CBVR) in large scale. CBVR is problem of retrieving most similar videos to a given query video or image.

'Content-based' means that search is done by analyzing the visual content rather than metadata (like tags or description) associated with it. Here, the term 'Content' means visual features extracted from the video data that describes and represents the video content. CBVR has wide range of applications such as video browsing, content filtering, video indexing, In-video advertising, and video surveillance. Over the past few decades considerable research progress has been made in Image Retrieval [1], however within the multimedia system community CBVR has not received enough attention. Conventional search techniques are difficult to use due to high computing costs when processing a large database of video clips. Hashing is a desirable solution to assist fast retrieval of videos in large-scale, by computing compact binary codes to represent video contents. Then similar videos can be search by using hamming distance. Hence, hashing based methods are computational efficient and require only limited storage space.

### *A. Convolutional Neural Network*

Convolutional Neural Network (CNN) is special kind of neural network from the field of deep learning specially design to analyze two dimensional visual data as a primary goal. It is network of large number of hidden layers mostly consist of convolutional layers. Such a network when trained with large dataset (consist of millions of images or videos), it learns to extract rich information present in the images or videos. CNN plays dominant role in solving the computer vision problems such as image classification, object detection, and others [2], and outperforms the previous methods based on hand engineered representations with large margin. In this paper we use resnet50 [3] as it shown balance between accuracy and efficiency in imagenet classification [4]. The network structure design of resnet50 is shown in figure 2.

### *B. Long Short Term Memory*

Recurrent neural networks (RNN) are a class of neural networks where previous step outputs are used as inputs to current step in the recurrent manner.

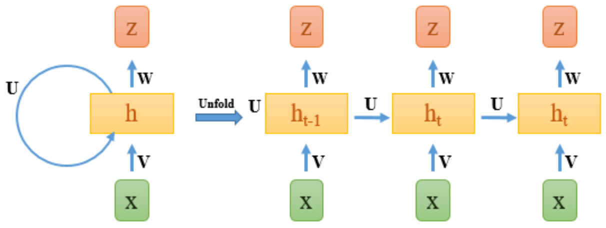

**Figure 1: Basic RNN model**

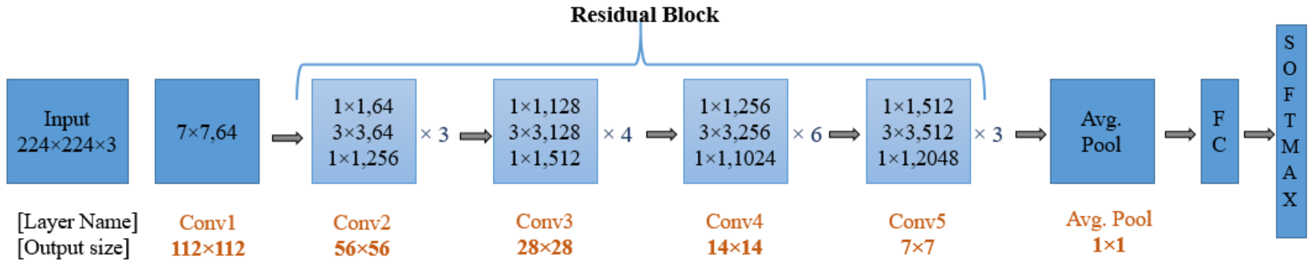

**Figure 2: Resnet 50 architecture, residual blocks means it consists of residual connection [4]. × a means number of blocks.**

Basic RNN as shown in figure 1 learns temporal dynamics in time sequence data by using following equations (1) and (2):

$$h_t = \omega\left(V * x_t + U * h_{t-1} + b_h\right) \quad (1)$$

$$z_t = \omega\left(W * h_t + b_z\right) \quad (2)$$

where $\omega$ is activation function (such as sigmoid), $x_t$ is the input, $h_t$ is the hidden state, $z_t$ is the output at time step t. W, U, V are weights and b is bias.

Training RNN to learn continuous dynamics in long-term is difficult because of vanishing and exploding gradients problem. To solve this problem Long short-term memory (LSTM) is developed in [5]. LSTM (figure 3) is recurrent modules which enables long-range learning with use memory cells and forget gate [5]. LSTM achieves superior performance on tasks such as machine translation and speech recognition [6].

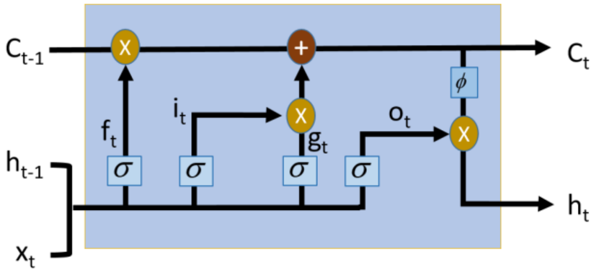

**Figure 3: LSTM model**

For a given input sequence $x_t$ at time $t$, following LSTM modules are updated for time step $t$ as follows:

$$i_t = \sigma\left(V_{xi} x_t + V_{hi} h_{t-1} + b_i\right) \quad (3)$$

$$f_t = \sigma\left(V_{xf} x_t + V_{hf} h_{t-1} + b_f\right) \quad (4)$$

$$g_t = \sigma\left(V_{xg} x_t + V_{hg} h_{t-1} + b_g\right) \quad (5)$$

$$o_t = \sigma\left(V_{xo} x_t + V_{ho} h_{t-1} + b_o\right) \quad (6)$$

$$c_t = \left(f_t \; e \; c_{t-1} + i_t \; e \; g_t\right) \quad (7)$$

$$h_t = o_t \; e \; \phi\left(c_t\right) \quad (8)$$

where V is a weight, b is bias, $\sigma$ is sigmoid activation function, $\phi$ is hyperbolic tangent function, e denotes element-wise multiplication and $i$, $f$, $g$, $o$, $c$ are input gate, forget gate, input modulation gate, output gate, cell (memory) activation vectors respectively. In this paper, we employ CNN and stacked LSTM for spatio-temporal representation of video. The spatial features of videos are extracted by CNN, and dynamic descriptors are built with stacked LSTM network. To obtain video-level representation time series pooling operation is done to pool the frame-level activations. To obtain a binary code for each video, video-level descriptors are fed into hashing component to learn its binary representation. So, our method exploits both the spatial and the temporal dynamics to build a compact video level representative binary code. This paper is structurize as: literature review is presented in Section II. Section III explains the proposed framework. Experimental settings are discussed in Section IV. Section V discusses the Results and discussion. Finally, Section VI summarizes the conclusion with future research directions.

## II. LITERATURE SURVEY

### *A. Hashing*

To approximate nearest neighbor search hashing is widely applied for large scale data. Gong et al. [7] presented Iterative quantization method to learn binary code, which rotates data point iteratively to minimize the binarization loss. In [8] spectral hashing is proposed in which by solving spectral graph partitioning problem binary codes are generated. In [9] the hash functions are estimated by thresholding the lower eigenfunctions of the Anchor Graph Laplacian in a hierarchical fashion. In [10] supervised video hashing framework is proposed which exploits both spatial and temporal information and learn hash function by minimizing structure-regularized empirical loss.

This method generates the codes at frame level. Cao et al. [11] proposed HashNet a deep learning architecture which learn binary codes by optimizing weighted pairwise cross-entropy loss function in deep CNNs by continuation method but ignore the temporal information. In [12] deep pairwise-supervised hashing method is proposed which simultaneous perform feature learning and hashcode learning with pairwise labels. Gu et al. [13] proposed a Supervised Recurrent Hashing that deploy LSTM to model the structural discriminative representation of video for designing a hashing function. Jingkuan Song et al. [14] proposed a multiple feature hashing technique which generates binary codes via multiple types of handcrafted features at frame levels.

### B. LSTM based Video Analysis

Video is a sequence of frames which contain temporal information and it is crucial for video content analysis. Ng et al. [15] proposed LSTM based sequence to sequence learning neural model for video classification.

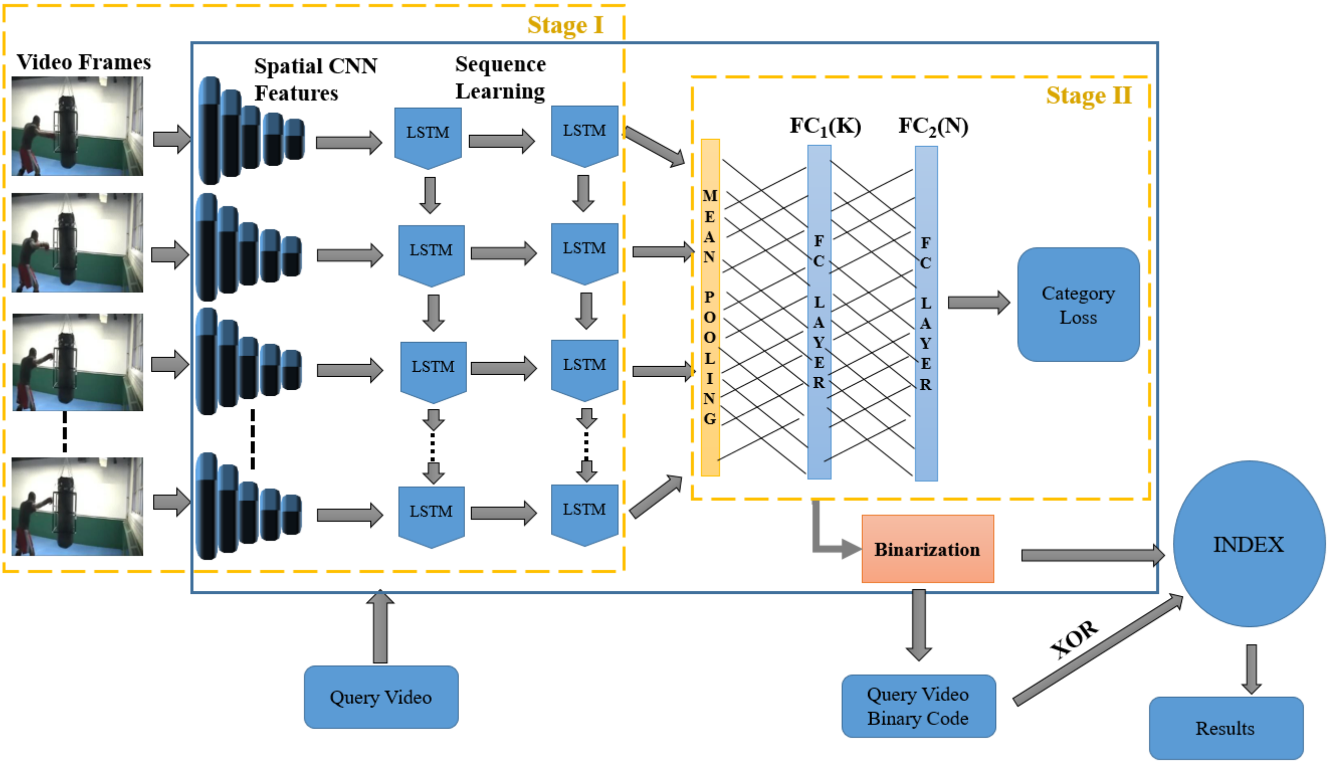

**Figure 4: Overview of the proposed framework. Stage I and II are training stages.**

Donahue et al. [17] exploits LSTM for video recognition and sequence to sequence captioning. [18] present a unsupervised learning mechanism of video representation is learn by using LSTM as encoder and decoder.

## III. PROPOSED FRAMEWORK

The outline of proposed framework is illustrate in figure 4. The overall procedure for our proposed CBVR system is divided into three core parts: First, set of consecutive frames of a video are fed to CNN to encode spatial information to rich features. Second, these features representing the sequence of action in video are input to the Stacked-LSTM to learn temporal dynamics in the video. Third, mean pooling is done over frame level LSTM responses to represent video level features followed by binary encoding to obtain binary code for retrieval task.

The first component of our method is CNN to transform high dimensional image to compact discriminative feature. For this we follow the resnet-50 architecture [3] which is pretrained on imagenet [4]. Resnet-50 (figure 2) takes a rgb image of spatial resolution 224 × 224 as input and pass through multiple residual blocks to average pool layer followed by FC layer and softmax. For spatial features we extract the output of average pooling layer in the resnet-50 as feature descriptor of 2048 dimensional by feeding frame to it. For each video $V_i$ , sequence of frame features $S_i$ are generated as follows:

$$S_i = f_{CNN}\left(V_i^{(t)}\right) \qquad (9)$$

where, t = 1,... $n_i$ (number of frames in $i^{th}$ video)

Now, the output { $S_i$ } from avgpool layer of CNN fed into the second component of proposed method i.e. LSTM as single time step at a time to generate hidden states $h_i^{l(t)}$ using (8). Having complex sequence patterns in large training data usually not identified by the single LSTM cell [15]. Therefore to learn long term dependencies in video sequence, Stacked-LSTM is used instead of single LSTM cell by stacking two LSTM cells of 1024 and 256 hidden units in the proposed approach.

The second LSTM layer's fame level responses $H_i = \left\{h_i^{2(1)},\ldots,h_i^{2(t-1)},h_i^{2(t)}\right\}$ of video $\mathrm{V_i}$ are integrated into fixed size of compact codes by mean-pooling layer the third component of proposed method as:

$$Zmean_i = f_{mean}(H_i) \qquad (10)$$

where $f_{mean}$ is mean pooling function.

After that k bit hash code $Hc_i$ of i$^{\text{th}}$ video is generated in fully connecter layer $\mathrm{FC_1}$ as follows:

$$Hc_i = \Psi\left(W * Zmean_i\right) \qquad (11)$$

where $*$ is matrix multiplication, $\mathrm{W}$ is weight matrix, $\Psi$ is thresholding operation to binarize the output of $(W * Zmean)$.

By training our network end to end including hashing does not converge on testing data and very slow in training data as shown in figure 5 and figure 6, this may be due to inclusion of low fully connected units $\mathrm{FC_1}$(K) (in figure 4) which forcing the model to transform data to low dimensional space which cause over fitting of network on data.

To overcome this, training of proposed framework is divided into two stages. In the First stage temporal dynamics of video are learn by keeping parameters of CNN freeze. In the second stage parameters of CNN and LSTM are freeze to learn compact real value code. For fast retrieval of videos hashing is commonly done by projecting high dimensional descriptors into binary space to obtain binary code. At the same time generated hash codes should also preserve sematic similarity. For that supervised cross-entropy loss is used in training stage 2.

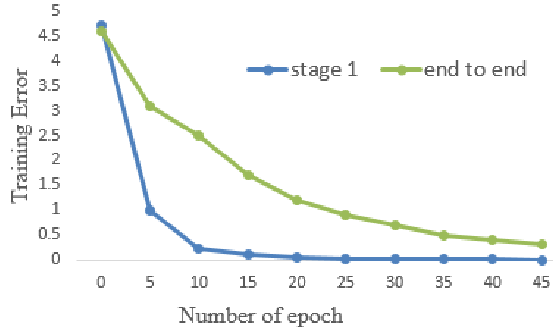

**Figure 5: training error**

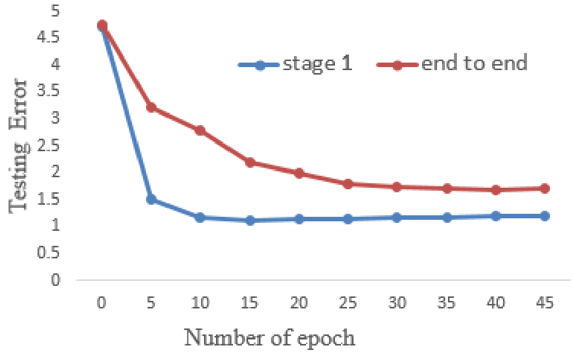

**Figure 6: testing error**

## IV. EXPERIMENTS

### A. Dataset and Setting

UCF-101 dataset [19] is chosen to conduct all the experiments. This dataset consists of approximate thirteen thousand videos having one-hundred-one categories. For training and testing we use the standard train/test split 1. Query is selected from testing set and retrieval is done from training set.

All the experiments are conducted using Matlab 2019a's deep learning toolbox along tesla k40c gpu for training our system.

### B. Implementation Details

Training stage 1: Each video is divided into clips of 16 frames with stride of 8 frames. Then random clip is selected from each video followed by resizing it to 256×256×16×3 and cropped randomly to 224×224×16×3 which inputs to the network. Adam optimizer is used to update the network parameters and minibatch size set to 64. Learning rate is set to .0001 for entire training.

Training stage 2: We use Scaled conjugate gradient back propagation algorithm [20] to update parameters of fully connected layer $\mathrm{FC_1}$ and $\mathrm{FC_2}$.

## V. EVALUATION MEASURES AND RESULTS

In this experiment some existing hashing methods are adopted as baselines including LSH [21], SpH [22], Spectral hashing (SH) [8], ITQ [7], AGH [9], PCAH [7], CBE [23], MFH [16] and carried out on the UCF101 dataset. For sake of fair comparison, video features are extracted from the same deep neural network and fed as inputs to these methods as in the proposed method rather than hand engineered features. Features from lstm and cnn are also extracted to test the performance of spatio-temporal and spatial features in retrieval of videos.

For evaluation of retrieval performance we adopted Mean Average Precision (MAP). MAP for all queries Q is computed using (12)

$$\mathrm{MAP} = \frac{1}{\mathrm{Q}}\sum_{\mathrm{q}}\left(\frac{1}{\mathrm{G_q}}\sum_{\mathrm{k=1}}^{\mathrm{R}}\mathrm{P_q}(\mathrm{k})\,\theta_{\mathrm{q}}(\mathrm{k})\right) \qquad (12)$$

where $\mathrm{G_q}$ is frequency of similar videos in the retrieved set for a query q. $\mathrm{P_q}(\mathrm{k})$ is the precision of the top k retrieved and $\theta$ is an sign function such that, $\theta_{\mathrm{q}}(\mathrm{k}) = 1$ if the item at rank k is a relevant, else 0.

**Table I.** MAP of different hashing methods using CNN based features methods with different code length

| **Methods** | **Binary Code Length** | | | | |
|---|---|---|---|---|---|
| | **8** | **16** | **32** | **64** | **128** |

| | | | | | |
|---|---|---|---|---|---|
| **CNN-AGH** | 0.083 | 0.111 | 0.165 | 0.230 | 0.261 |
| **CNN-SH** | 0.075 | 0.102 | 0.151 | 0.189 | 0.211 |
| **CNN-LSH** | 0.032 | 0.034 | 0.060 | 0.097 | 0.155 |
| **CNN-PCAH** | 0.071 | 0.091 | 0.117 | 0.139 | 0.142 |
| **CNN-SpH** | 0.061 | 0.088 | 0.130 | 0.175 | 0.219 |
| **CNN-CBE** | 0.034 | 0.034 | 0.034 | 0.034 | 0.031 |
| **CNN-CBE** | 0.034 | 0.034 | 0.034 | 0.034 | 0.031 |
| **CNN-MFH** | 0.076 | 0.110 | 0.167 | 0.225 | 0.264 |
| **CNN-ITQ** | 0.075 | 0.122 | 0.169 | 0.227 | 0.269 |
| **Proposed Method** | **0.540** | **0.649** | **0.681** | **0.698** | **0.705** |

MAP of all the methods including proposed work are reported in table I and table II. From table 1 it is clear that proposed method is far ahead of other methods. In table 2 when temporal information is used by all other hashing methods retrieval performance improves over that of CNN based spatial feature which and demonstrate the power of temporal information present in the video.

Proposed method is slightly less than LSTM-ITQ in bit code length of 32, 64,128. But as bit length becomes shorter proposed method performs superiorly than other. So our method represents the video content with less information loss compared to others at low bit binary code.

**Table II. MAP of different hashing methods using LSTM based features methods with different code length**

| **Methods** | **Binary Code Length** | | | | |
|---|---|---|---|---|---|
| | **8** | **16** | **32** | **64** | **128** |
| **LSTM-AGH** | 0.294 | 0.448 | 0.563 | 0.617 | 0.684 |
| **LSTM-SH** | 0.410 | 0.599 | 0.664 | 0.684 | 0.686 |
| **LSTM-LSH** | 0.275 | 0.493 | 0.585 | 0.660 | 0.694 |
| **LSTM-PCAH** | 0.413 | 0.603 | 0.660 | 0.679 | 0.649 |
| **LSTM-SpH** | 0.234 | 0.387 | 0.507 | 0.562 | 0.592 |
| **LSTM-CBE** | 0.316 | 0.305 | 0.551 | 0.564 | 0.590 |
| **LSTM-MFH** | 0.421 | 0.591 | 0.661 | 0.697 | 0.703 |
| **LSTM-ITQ** | 0.450 | 0.638 | **0.695** | **0.718** | **0.727** |
| **Proposed Method** | **0.540** | **0.649** | 0.681 | 0.698 | 0.705 |

## VI. CONCLUSION AND FUTURE WORK

This paper presents a framework of fast content based video retrieval via deep learning with hashing. Specifically, deep CNN along with LSTM is deployed to learn spatio-temporal representations of video. For fast retrieval of videos the video is encoded as binary code by hashing layer on the basis of learned spatio-temporal features. For effective learning of hash function, whole framework is trained in two stages. Stage I learn the temporal context of videos. Stage II learns to transform the features learn from Stage I into a compact dimensional space followed by hashing. Results states that the proposed approach is able to represent video content with less information loss at low bits 8, 16 as compared to other hashing methods. Future work includes use of optical flow or other motion information along with training in large dataset which can improve the retrieval performance.

## REFERENCES

1. Zhou, Wengang, Houqiang Li, and Qi Tian. "Recent advance in content-based image retrieval: A literature survey." *arXiv preprint arXiv:1706.06064* (2017).
2. Guo, Yanming, Yu Liu, Ard Oerlemans, Songyang Lao, Song Wu, and Michael S. Lew. "Deep learning for visual understanding: A review." *Neurocomputing* 187 (2016): 27-48.
3. He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778. 2016.
4. Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang et al. "Imagenet large scale visual recognition challenge." *International journal of computer vision* 115, no. 3 (2015): 211-252.
5. Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." *Neural computation* 9, no. 8 (1997): 1735-1780.
6. T. Young, D. Hazarika, S. Poria and E. Cambria, "Recent Trends in Deep Learning Based Natural Language Processing [Review Article]," in *IEEE Computational Intelligence Magazine*, vol. 13, no. 3, pp. 55-75, Aug. 2018.
7. Gong, Yunchao, Svetlana Lazebnik, Albert Gordo, and Florent Perronnin. "Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, no. 12 (2012): 2916-2929.
8. Weiss, Yair, Antonio Torralba, and Rob Fergus. "Spectral hashing." In *Advances in neural information processing systems*, pp. 1753-1760. 2009.
9. Liu, Wei, Jun Wang, Sanjiv Kumar, and Shih-Fu Chang. "Hashing with graphs." In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pp. 1-8. Omnipress, 2011.
10. Ye, Guangnan, Dong Liu, Jun Wang, and Shih-Fu Chang. "Large-scale video hashing via structure learning." In *Proceedings of the IEEE international conference on computer vision*, pp. 2272-2279. 2013.
11. Cao, Zhangjie, Mingsheng Long, Jianmin Wang, and Philip S. Yu. "Hashnet: Deep learning to hash by continuation." In *Proceedings of the IEEE international conference on computer vision*, pp. 5608-5617. 2017.
12. Li, Wu-Jun, Sheng Wang, and Wang-Cheng Kang. "Feature learning based deep supervised hashing with pairwise labels." *arXiv preprint arXiv:1511.03855* (2015).
13. Gu, Yun, Chao Ma, and Jie Yang. "Supervised recurrent hashing for large scale video retrieval." In *Proceedings of the 24th ACM international conference on Multimedia*, pp. 272-276. ACM, 2016.
14. Song, Jingkuan, Yi Yang, Zi Huang, Heng Tao Shen, and Richang Hong. "Multiple feature hashing for real-time large scale near-duplicate video retrieval." In *Proceedings of the 19th ACM international conference on Multimedia*, pp. 423-432. ACM, 2011.
15. Yue-Hei Ng, Joe, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. "Beyond short snippets: Deep networks for video classification." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4694-4702. 2015.
16. Venugopalan, Subhashini, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. "Sequence to sequence-video to text." In *Proceedings of the IEEE international conference on computer vision*, pp. 4534-4542. 2015.
17. Donahue, Jeffrey, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. "Long-term recurrent convolutional networks for visual recognition and description." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2625-2634. 2015.
18. Srivastava, Nitish, Elman Mansimov, and Ruslan Salakhudinov. "Unsupervised learning of video representations using lstms." In *International conference on machine learning*, pp. 843-852. 2015

19. Soomro, Khurram, Amir Roshan Zamir, and Mubarak Shah. "UCF101: A dataset of 101 human actions classes from videos in the wild." *arXiv preprint arXiv:1212.0402* (2012)
20. Møller, Martin Fodslette. "A scaled conjugate gradient algorithm for fast supervised learning." *Neural networks* 6, no. 4 (1993): 525-533.
21. Gionis, Aristides, Piotr Indyk, and Rajeev Motwani. "Similarity search in high dimensions via hashing." In *Vldb*, vol. 99, no. 6, pp. 518-529. 1999.
22. Heo, Jae-Pil, Youngwoon Lee, Junfeng He, Shih-Fu Chang, and Sung-Eui Yoon. "Spherical hashing." In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2957-2964. IEEE, 2012.
23. Yu, Felix, Sanjiv Kumar, Yunchao Gong, and Shih-Fu Chang. "Circulant binary embedding." In *International conference on machine learning*, pp. 946-954. 2014.

## AUTHORS PROFILE

**Mr. Vidit Kumar** has done B.Tech in Computer Science and Engineering from Uttaranchal University, M.Tech from Graphic Era deemed to be university, and currently pursuing PhD in Computer Science and Engineering from Graphic Era deemed to be university, Dehradun, India. His research of interest is in Machine Learning, Deep Learning, Video Analytics and Computer Vision.

**Dr. Vikas Tripathi** has done BE in information technology from Technocrats institute of technology, Bhopal, M. Tech in Software engineering from Indian institute of information technology Gwalior and PhD from Uttarakhand technical university, Dehradun. He is actively involved in research related to Software engineering, Computer Vision, Machine learning and Video Analytics.
He has published many papers in reputed international conferences and journals. Currently he is working as an associate professor in Graphic era deemed to be university Dehradun, India.

**Dr. Bhasker Pant** Currently working as Dean Research & Development and Associate Professor in Department of Computer Science and Engineering. He is Ph.D. in Machine Learning and Bioinformatics from MANIT, Bhopal. Has more than 15 years of experience in Research and Academics. He has till now guided as Supervisor 3 Ph.D. candidates (Awarded).and 5 candidates are in advance state of work. He has also guided 28 MTech. Students for dissertation. He has also supervised 2 foreign students for internship. Dr.Bhasker Pant has more than 70 research publication in National and international Journals. He has also chaired a session in Robust Classification & Predictive Modelling for classification held at Huangshi, China.