Identification of Visual Objects in Lecture Videos with Color and Keypoints Analysis

Dipayan Biswas Shishir Shah Jaspal Subhlok
Department of Computer Science, University of Houston, Houston, TX
Email: dbiswas@uh.edu, sshah@central.uh.edu, jaspal@uh.edu

I. ABSTRACT

***Abstract*—Recorded lecture videos are an increasingly important learning resource. However, traditional video format does not allow quick navigation to the desired content of interest. Recent research has enhanced navigation by dividing lecture videos into chapters and creating a summary of each chapter. The visual content on lecture video frames represents a valuable source of information for identifying topic boundaries as well as summarizing content. The focus of the research presented in this paper is to accurately identify visual objects in lecture video frames. The methods developed for camera videos are not directly applicable here as the visual content includes charts, graphs, and illustrations intermingled with text. A common approach based on locating regions with continuous pixel changes has a key limitation that logically consistent visual objects can have modest size gaps inside them. The result is over-segmentation, where a logical object is split into multiple objects if the gap threshold is too low, or under-segmentation, where adjacent objects are recognized as a single large object if the gap threshold is too high. This paper introduces a novel approach that exploits the observation that components of logical objects often have color and geometrical similarity. In our methodology, first a relatively large number of visual elements are identified with a small gap threshold. Subsequently, these visual elements are selectively combined using gap along with color and geometrical similarity. An evaluation was conducted with a suite of 170 lecture video frames from STEM coursework. The results demonstrate the significant impact of color and geometry in improving the accuracy of visual object identification in lecture video frames.**

II. INTRODUCTION

Recorded lecture video is an essential learning resource that complements a conventional live lecture [1]. Videos provide learners an opportunity to access lecture content anytime and anywhere. Students can employ video to recover from a missed class or to review challenging topics. Studies show that students take advantage of recorded lectures to prepare for exams and tests, and they can have a positive impact on overall grades [2].

The main limitation of the traditional video format is the lack of quick access to the content of interest. Scrolling back and forth to find the desired content in a long lecture video is time-consuming and limits the usage potential. Lecture videos lack indexing capabilities akin to a table of contents or a list of tables and figures in a textbook.

This research has its roots in the *VideoPoints* project, an advanced lecture video platform to improve navigation inside a lecture video, available at *www.videopoints.org*. Videopoints presents a lecture video as a sequence of topical segments with a text and visual summary of each subtopic segment [3], [4] as illustrated in Figure 1. Users can quickly navigate to the content of interest by viewing summaries of different topic segments. Currently, topic transitions in Videopoints are identified by analysis of screen text. However, we noticed several instances where image similarity analysis would have improved results.

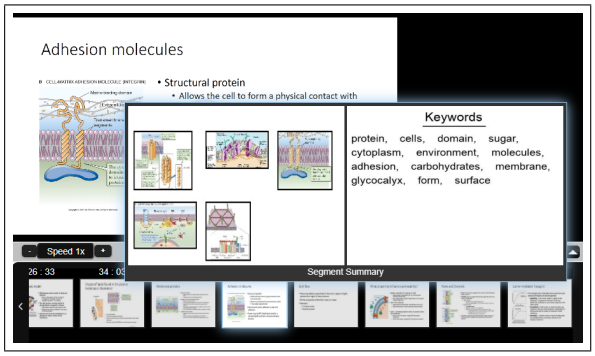


Fig. 1. Videopoints player showing topical indexing and summaries

It is also clear that the visual content in a lecture video is important information for improving navigation. *The objective of the research presented in this paper is to accurately identify visual objects in lecture video frames.* The results will be applied to improve lecture video navigation with more accurate topic transitions and improved summaries.

Lecture video frames are different from movie video frames as they contain charts, graphs, tables, illustrations, and photographs. The traditional approach to identifying visual objects on a video frame is to locate regions of pixel intensity changes with a sliding window, which assumes that blank space separates visual objects. This approach has significant limitations that became clear in the context of previous research [5] and poses a unique challenge. The fundamental issue is that an author's intended logical illustration may not align with a contiguous set of physical drawings reflected by pixel changes. For instance, a flow chart may have gaps between steps, although it is a single logical object. At the same time, an author may place multiple visual objects close to each other that may get recognized as a single object.

The insight that motivated this project is that nearby visual entities are more likely to be part of the same visual object if they share other attributes, specifically *color* and *local geometry*. This paper investigates the role of these factors

in the identification of visual objects in lecture videos. Correlation between color histograms is used to measure color similarity, and keypoints similarity estimates commonality in local geometry.

This paper introduces a new algorithm named *Logiform* that combines color histograms, keypoints, and pixel intensity changes for visual object identification. The algorithm was implemented and evaluated on a suite of lecture video frames from STEM coursework. Results show that color and local geometry can play an important role in improving the performance of visual object detection.

III. Related Work

The broad motivation for this research is to improve the navigation of lecture videos. Active research in this direction focuses on dividing a lecture video into segments covering subtopics [3], [6]–[8]. Some approaches identify topic transition points using speech text and/or screen text extracted from lecture video frames. Other efforts focus on indexing or summarization of lecture video content in a variety of ways. They span selecting important keyframes based on *i)* historical user interaction [9], *ii)* amount of textual content [10], or *iii)* amount of visual content and display duration [11]. In our prior work, we have developed summaries of lecture videos containing keywords and important images [4]. Research presented in this paper aims to accurately identify individual visual objects, which in turn can have a positive impact on enhancing this direction of research.

The general problem of object detection in videos has been an active area of research for decades in the context of a number of applications, including autonomous vehicles, surveillance, industrial automation, and robotics [12]. However, identifying visual objects in a lecture video frame is a unique problem. Common challenges for general object detection tasks such as illuminations, occlusion, shadows, and complex backgrounds do not apply to the lecture video domain. Common visual objects on lecture video frames are charts, graphs, and illustrations, not camera images. Methods for feature extraction in traditional object detection can be employed to identify fine-grain visual elements, but they are not sufficient to address the challenge of identifying meaningful visual objects created by the authors of video content.

The visual object detection on the lecture video frames is a relatively under-explored area. ViZig [13] identifies the location of "anchor points" in lecture video frames that can be figures, tables, equations, flowcharts, code snippets, and charts. They formulated a classification problem that employed a deep convolution neural network using unconstrained internet images. However, this work does not address identifying the location or details of these objects that are needed for tasks like topic discovery. In a related project [14], they associate a text description with an extracted visual element. Their method for extracting visual elements groups together neighbors based on factors like the height and centroid of their bounding boxes. Our research explores additional features including color and geometry, and is driven by diverse STEM coursework.

IV. Motivation: Color and Geometry in Visual Content Identification

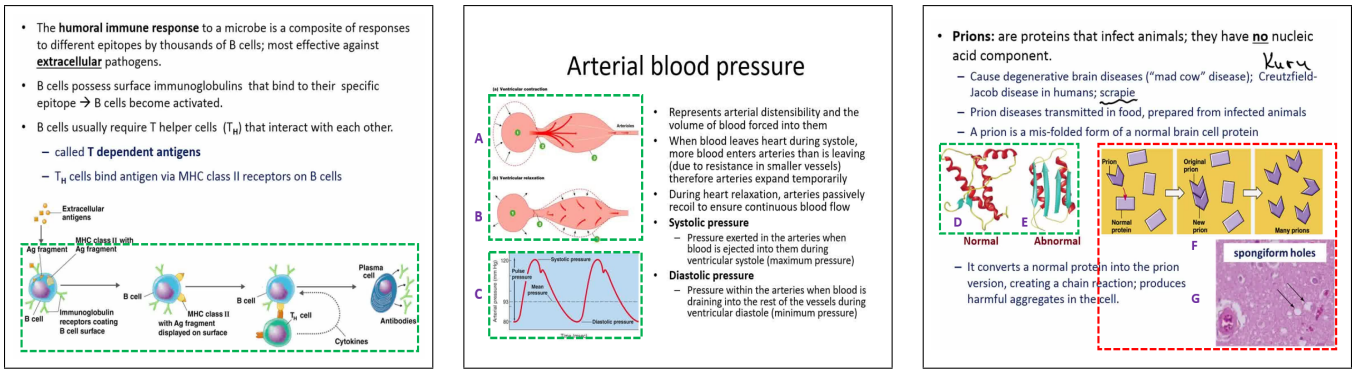
Algorithms for identifying visual objects that rely on the assumption that a visual object is surrounded by clear space or a gap are constrained to the amount of separation (clear space) between them. We have discovered that if this gap threshold is too small, then a composite figure consisting of multiple components is often identified incorrectly as multiple objects, which is referred to as over-segmentation. On the other hand, a large gap threshold will often identify multiple disparate images as a single object. Hence, it is important to capture semantic or logical relationships between image components in the context of lecture video frames.

We present some example video frames that motivated this research. Text boxes are identified and replaced by blank space prior to visual object analysis but retained here for context. In the following figures, dashed green boxes represent the ground truth visual objects in the frames.

Figure 2(a) consists of several small images that are components of a logical flowchart. A basic spatial gap based algorithm may incorrectly identify individual images as visual objects because of the space separation between them. Figure 2(b) consists of three visual sub-objects marked A, B, and C. The author's intent and human intuition is that A and B are components of a single visual object, while C is a graph that is clearly a separate entity. However, a space gap based algorithm is likely to keep A and B separate because of the significant gap between them, and may instead combine B and C based on the small gap. Finally, Figure 2(c) consists of 4 image sub-objects labeled D, E, F, and G. Clearly, D and E are components of a single logical visual object, but a gap based algorithm may identify them as separate objects. On the other hand, F and G are intended to be separate objects, but a gap based algorithm is likely to identify them as a combined object because of the very small gap between them.

The point is that relying entirely on spatial separation to identify and separate visual objects is error-prone. We now focus on the color and geometry aspects of the visual objects in these figures. In Figure 2(a), small images that are components of the flowchart have similar color and geometric properties. In Figure 2(b), sub-objects A and B have strong geometric and color similarities between them, but no similarity with C. Finally, in Figure 2(c), there is a strong color and geometric similarity between sub-objects D and E, while they have little similarity to any other visual sub-objects. Sub-objects F and G have a very small gap between them but a low degree of color or geometric similarity. We skip details for brevity, but it is clear that considering color and geometric similarity can lead to improved object identification in these examples.

In summary, space separation based visual object detection faces significant challenges in the context of lecture video frames. At the same time, components of a visual object often share color and geometric similarities. This paper explores how these similarities can be employed to enhance the accuracy of visual object identification in lecture video frames.



(a) (b) (c)

Fig. 2. Sample lecture video frames to highlight the limitations of a gap based approach to visual object identification

V. Visual Object Identification

We present a framework for identifying visual objects in a lecture video frame. The goal is to identify parts of a video frame that constitute logically meaningful objects to developers and consumers of lecture videos, not just physically contiguous pixels. The innovation is employing color and geometrical similarity in addition to local pixel variance in locating the boundaries of visual objects.

We refer to our proposed method as the *LogiForm* algorithm. The input to the algorithm is a lecture video frame with RGB pixel values, and the output is the coordinates of a set of rectangles that represent the bounding boxes of the visual objects. In the preprocessing steps for this algorithm, the video frame is resized to Standard Definition resolution if needed, and all text regions are identified with Optical Character Recognition [15] and masked. Hence, those regions appear as white space with no pixel variation. The algorithm consists of three major steps as depicted in Figure 3. They are outlined below and then covered in more detail.

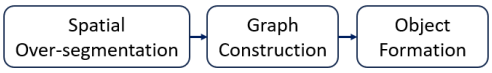


Fig. 3. Steps of the LogiForm algorithm

1) *Spatial Oversegmentation:* A simple sliding window technique is employed to detect contiguous regions with pixel variations surrounded by gaps. The parameters are set to detect a larger number of regions than the likely number of actual visual objects.
2) *Graph Construction:* The visual object regions identified become graph nodes. Edges between graph nodes are assigned weights based on the Euclidean distance as well as color and local geometry (dis)similarity.
3) *Object formation:* A hierarchical clustering algorithm is applied that selectively merges graph nodes based on the edge weights, yielding the final visual objects.

In the post-processing phase, some visual objects are removed based on real-world considerations. These include objects that are too small or too large or have a very large or small height to width ratio. We now detail the three main steps in visual object identification.

A. Spatial Oversegmentation

A video frame is segmented into regions based on the variation in pixel intensities, with a threshold selected to favor over-segmentation. First, the pre-processed image is converted to a grayscale image. Then, we calculate the local variance at each pixel compared to neighbors using a sliding window technique. Thresholding is applied to recognize foreground pixels based on their high variance. A blob coloring algorithm using 8-connectivity neighborhood is applied to combine high variance pixels in the thresholded binary image into connected regions. The result is a set of regions, each representing an area of visual content on the lecture video frame.

Sliding window size is a key parameter in this segmentation step. A large window size is likely to lead to *undersegmentation* where some neighboring visual entities may be identified together as a single visual object, while a small window size is likely to lead to *oversegmentation* where a meaningful visual object with small space gaps inside may be recognized as multiple visual entities. In our experiments with a set of 170 lecture video frames, the number of regions identified varied from 598 for a window size of 3x3 to 452 for a window size of 15x15. The best performance was obtained for a 9x9 window size. The approach used in this paper aims to over-segment with a larger number of smaller visual objects in this step and employ a more sophisticated approach, including color and geometry considerations, to selectively combine the regions into larger visual objects. For the experiments presented in this work, a window size of 5x5 was heuristically selected.

B. Graph Construction

The set of regions identified with pixel intensity-based analysis form the nodes of this graph. The graph edges have three attributes:

- Minimum gap (*minGap*) representing the shortest Euclidian distance between the pairs of nodes representing

regions. The distance value is normalized to a 0-1 range after accounting for image resolution and frame size.

- Color difference (*colorDiff*) representing the difference in color space between a pair of regions. We compute color histograms for all node regions using the pixel values of the three color channels of the input frame. Then color similarity is calculated based on the correlation score between pairs of color histograms. Finally, we transform the correlation score to a color dissimilarity score *colorDiff* in the 0-1 range to align with the graph representation where a larger score reflects more distance.[1]
- Keypoints dissimilarity (*KPDiss*) captures the (lack of) local geometric similarity between region nodes connected by the corresponding graph edge based on SIFT feature descriptors [16] computed on the grayscale frame for the corresponding regions. The similarity between pairs of nodes is calculated by the keypoints match ratio between the SIFT descriptors. The similarity score is transformed to a keypoints dissimilarity score *KPDiss* in the 0-1 range to align with other graph edge attributes.

The final graph edge cost is a linear combination of the three cost components represented as follows:

$$edgeCost = minGap * gapWeight + colorDiff * colorWeight + KPDiss * KPWeight \quad (1)$$

where $gapWeight$, $colorWeight$ and $KPWeight$ are heuristically determined parameters.

C. Object Formation

In this step, hierarchical clustering is applied to the constructed graph to selectively combine nodes based on graph edges representing minimum gap, color, and keypoint similarities. A bottom-up agglomerative clustering algorithm [17] was employed using the *scikit-learn* package. Initially, each node is considered an individual cluster, and pairs of clusters are combined recursively. A key consideration in agglomerative clustering is *cluster linkage* criteria that decides the use of edge weights to cluster nodes that are formed by combining multiple cluster nodes. We take the minimum of the edge weights as the weight of the new edge formed after combining, which is a natural choice due to the Euclidean nature of the problem. Another key parameter is the *merge Threshold*: clusters are combined in each iteration where the linkage (or *edgeCost*) is below the *merge Threshold*. A higher value of *merge Threshold* results in more merges and fewer clusters and vice-versa. We will report on experiments with different values of merge threshold. The set of clusters after this step represents the final image objects.

VI. Evaluation and Results

The LogiForm algorithm discussed in Section V was implemented and evaluated in the context of the Videopoints lecture video platform [3]. We describe the data set and metrics used for evaluation and present results.

[1]Order preserving nonlinear transformations were used to account for the skewed distribution of raw colorDiff and KPDiss scores.

A. Dataset

We have access to a large set of lecture videos from earlier research conducted on the Videopoints platform [3], [4]. We selected a suite of 170 video frames from 53 lecture videos in Biology, Geosciences, and Computer Science. The primary criterion for the selection of a lecture video frame for this study is the presence of significant visual content. Ground truth boundaries for visual objects on frames were provided by volunteers familiar with the video content using *LabelImg*, a publicly available marking tool [18].

B. Evaluation Methodology

The visual object identification framework generates the coordinates for a set of bounding boxes for detected visual objects. For evaluation, these are compared against the ground truth set of bounding boxes. In practice, a 100% match between the two sets of bounding boxes is unlikely, and partial matches are important. This is a key consideration in selecting an evaluation metric. The evaluation was primarily done with mean Average Precision (mAP), a commonly used metric in object detection tasks in computer vision. The *mAP* is defined as the mean of average precision (*AP*) for all the classes, where *AP* is calculated by the area under the precision-recall curve for each class. A partial match threshold called Intersection over Union (IoU) of the two boxes to decide on a positive match was set to 0.5, a standard practice for object detection tasks. Measurements were made with an open-source toolkit [19]. A set of experiments was conducted by varying parameters in our Logiform object identification framework:

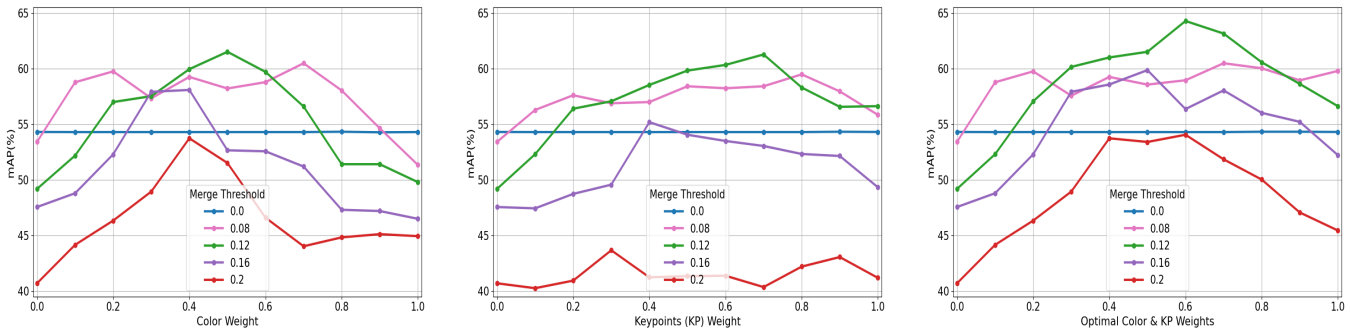
- *Merge Threshold* was varied from 0.0, representing no combining of regions in the object formation stage to 0.2, representing the least constraints on combining regions.
- For each value of the merge threshold, several sets of values of the parameters $gapWeight$, $colorWeight$, and $KPWeight$ were selected such that their sum equals 1.

We present results on the individual impact of color and local geometry on performance, followed by results on the combined impact of color and geometry.

C. Impact of Color

Figure 4(a) plots the performance of visual object identification with different color weights. A higher color weight means that color is given more consideration in combining nearby visual entities than the gap between them. Each curve represents a different *merge threshold* implying more merges during object formation, as discussed in section V. A merge threshold of 0 is represented by the flat blue line in the figure. In this scenario, the spatially segmented visual entities are the final detected objects, and color has no impact. For the remaining curves, a bell-shaped pattern is observed. As the color weight increases, performance improves, then reaches a peak, and starts to drop[2]. A high merge constraint of 0.2 (red line) leads to overall poor performance, although it is improved

[2]Minor variations to general patterns are expected statistical anomalies in the curves in this section due to the modest size of our dataset.



(a) (b) (c)

Fig. 4. Impact of combining (a) color similarity based on color histogram, (b) local geometric similarity based on SIFT keypoints, and (c) the best combination of color & local geometric similarity, with the gap in identifying visual objects. A lower/higher merge threshold leads to less/more merging of visual entities during object formation.

with midrange color weight. The best performance is observed with a merge threshold of 0.12 (green line) with a color weight of 0.5. The conclusion is that combining color similarity with the gap between segmented visual entities plays a significant positive role in accurately identifying visual objects.

D. Impact of Local Geometry

Figure 4(b) plots the performance of visual object identification with different levels of consideration to local geometric similarity based on keypoints match as detailed in Section V. For low to moderate values of merge threshold (green and pink curves), the impact of geometry is similar to that of color, although the best performance is at higher values of keypoints weight, and the drop in performance is more moderate as the keypoints weight is increased to 1. Also, the performance with the highest merge threshold of 0.2 (red curve) does not improve meaningfully with geometry considerations. The best performance is again observed for a merge threshold of 0.12 (green curve) but with keypoints weight of 0.7. Similar to color, the main conclusion is that combining local geometry with the gap between segmented visual entities improves the accuracy of identifying visual objects.

E. Combined Impact of Color and Geometry

We explore the impact of combining color, local geometry, and gap on identifying visual objects. Figure 4(c) shows experimental results for the optimal combination of color and local geometry with gap. In this figure, the x-axis represents the *experimentally measured optimal contribution of color and local geometry at each point.* (Say the gap weight is set to 0.4. For all combinations that add up to 0.6, if color weight = 0.4 and keypoints weight = 0.2 yields the best results, then that value is plotted.) The results are similar to those with color alone presented in Figure 4(a). However, the peak performance obtained with merge threshold = 0.12 (green curve) is moderately higher than the best performance with color or geometry alone, implying that combining color and local geometry provides benefits beyond using them individually.

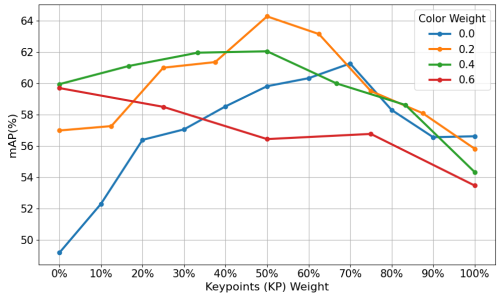


Fig. 5. Impact of SIFT keypoints based on local geometric similarity over different fixed color weights.

We provide another view of the results combining color, local geometry, and gap in Figure 5 but skip the details for brevity. Here the merge threshold is set to 0.12, the optimal value based on our experiments. Each curve represents a fixed color weight with the remaining weight divided between gap weight and keypoints weight such that the total is 1.0. $gapWeight + colorWeight + KPWeight = 1.0$. We note that the best results are for color weight = 0.2 with 50% of the remaining weight assigned as keypoints weight. Also, for the highest color weight of 0.6 (red line), keypoints consideration does not improve performance.

F. Summary of Results

The key result from our experiments is that consideration of color and local geometry leads to significant enhancement in the performance of visual object identification in lecture video frames, and optimal performance requires combining them with spatial considerations. A summary of results in the form of Precision, Recall, and mAP scores is presented in Table I. The best performance for a traditional pixel variance based segmentation algorithm was 55.52% mAP obtained with a window size of 9x9. The performance with our Logiform algorithm using color or local geometry independently reaches 61.59% mAP and 62.30% mAP, respectively. The performance achieved using color and local geometry together is 64.28% mAP representing a 15.8% improvement over the best perfor-

mance with a traditional pixel variance based method achieved with a 9x9 window size on our dataset. While the performance of visual object identification is far from perfect, it is important to note that humans differ considerably in precisely identifying visual objects in this domain, thereby limiting the best possible performance against a human generated ground truth.

TABLE I
PERFORMANCE OF VISUAL OBJECT IDENTIFICATION ON LECTURE VIDEOS

Experiments	**Precision(%)**	**Recall(%)**	**mAP(%)**
Segmentation with 9*9	72.39	74.39	55.52
Segmentation with 5*5	73.24	71.97	54.56
Logiform (gap + color)	78.57	76.12	61.51
Logiform (gap + geometry)	**79.78**	75.09	62.30
Logiform (gap + color + geometry)	78.89	**78.89**	**64.28**

VII. CONCLUSIONS AND FUTURE WORK

Visual content plays an important role in improving navigation and usability of lecture videos. Applications include dividing a lecture video into topical segments and generating summaries of the segments. Identifying visual content accurately on lecture video frames is central to these applications. However, traditional spatial pixel variance based methods have limited success in this domain as a lecture video frame is an unstructured document, and the human perception of what constitutes a logical and meaningful visual entity is different from the algorithmic approach of using pixel level variance to form the boundaries of visual objects.

This paper introduces Logiform algorithm, which we believe is the first work that combines pixel level variance with color and geometry for identifying visual objects in lecture video frames. A key problem is that a small spatial gap can be the divider between a pair of visual objects, or an empty space inside an object. The hypothesis that we test is that visual entities separated by a small space that have color and/or geometrical similarities are more likely to be parts of the same visual object. Our results show a meaningful 15.8% mAP improvement with this approach on a data set of 170 frames from STEM lecture videos with significant visual content.

We believe this paper opens a novel research direction in forming complete visual objects for lecture video frames with properties that bridge the gap between logical or human understanding of visual objects and machine understanding of pixel variations. Future work will extend the ground truth to a larger and more diverse set of lecture videos. A larger body of ground truth will also allow us to go beyond heuristics for best use of color and geometry information and explore machine learning based approaches to this problem. We also plan to study the real-world impact of improved video content detection by incorporating them into frameworks for indexing lecture videos and summarization of video content that are part of the Videopoints advanced lecture video environment.

ACKNOWLEDGEMENT

The authors wish to express their sincere gratitude to all current and former members of the Videopoints team, especially Mohammad Rajiur Rahman who laid the foundation for this work and Jatindera Walia who manages the frameworks employed. Partial support was received from the National Science Foundation under award NSF-SBIR-1820045.

REFERENCES

[1] L. Barker, C. L. Hovey, J. Subhlok, and T. Tuna, "Student perceptions of indexed, searchable videos of faculty lectures," in *Proceedings of the 44th Annual Frontiers in Education Conference(FIE)*, Madrid, Spain, Oct 2014.

[2] P. E. Dickson, D. I. Warshow, A. C. Goebel, C. C. Roache, and W. R. Adrion, "Student reactions to classroom lecture capture," in *Proceedings of the 17th ACM Annual Conference on Innovation and Technology in Computer Science Education*. NY, USA: ACM, 2012, p. 144–149.

[3] T. Tuna, J. Subhlok, L. Barker, S. Shah, O. Johnson, and C. Hovey, "Indexed captioned searchable videos: A learning companion for STEM coursework," *Journal of Science Education and Technology*, vol. 26, no. 1, pp. 82–99, 2017.

[4] M. R. Rahman, R. S. Koka, S. K. Shah, T. Solorio, and J. Subhlok, "Enhancing lecture video navigation with AI generated summaries," *Education and Information Technologies*, pp. 1–24, 2023.

[5] M. R. Rahman, "Visual summarization of lecture videos to enhance navigation," Ph.D. dissertation, Department of Computer Science, University of Houston, May 2021.

[6] A. Biswas, A. Gandhi, and O. Deshmukh, "Mmtoc: A multimodal method for table of content creation in educational videos," in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 621–630.

[7] P. A. Co, W. R. Dacuyan, J. G. Kandt, S.-C. Cheng *et al.*, "Automatic topic-based lecture video segmentation," in *International Conference on Innovative Technologies and Learning*. Springer, 2022, pp. 33–42.

[8] M. Furini, S. Mirri, and M. Montangero, "Topic-based playlist to improve video lecture accessibility," in *2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 2018, pp. 1–5.

[9] J. Kim, P. J. Guo, C. J. Cai, S.-W. Li, K. Z. Gajos, and R. C. Miller, "Data-driven interaction techniques for improving navigation of educational videos," in *Proceedings of the 27th annual ACM symposium on User interface software and technology*, 2014, pp. 563–572.

[10] K. Yadav, K. Shrivastava, S. Mohana Prasad, H. Arsikere, S. Patil, R. Kumar, and O. Deshmukh, "Content-driven multi-modal techniques for non-linear video navigation," in *Proceedings of the 20th international conference on intelligent user interfaces*, 2015, pp. 333–344.

[11] B. Zhao, S. Lin, X. Qi, R. Wang, and X. Luo, "A novel approach to automatic detection of presentation slides in educational videos," *Neural Computing and Applications*, vol. 29, pp. 1369–1382, 2018.

[12] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proceedings of the IEEE*, 2023.

[13] K. Yadav, A. Gandhi, A. Biswas, K. Shrivastava, S. Srivastava, and O. Deshmukh, "Vizig: Anchor points based non-linear navigation and summarization in educational videos," in *Proceedings of the 21st International Conference on Intelligent User Interfaces*, 2016, pp. 407–418.

[14] C. Xu, R. Wang, S. Lin, X. Luo, B. Zhao, L. Shao, and M. Hu, "Lecture2note: Automatic generation of lecture notes from slide-based educational videos," in *2019 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2019, pp. 898–903.

[15] A. Kay, "Tesseract: an open-source optical character recognition engine," *Linux Journal*, vol. 2007, no. 159, p. 2, 2007.

[16] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, pp. 91–110, 2004.

[17] D. Müllner, "Modern hierarchical, agglomerative clustering algorithms," *arXiv preprint arXiv:1109.2378*, 2011.

[18] Tzutalin, "Labelimg. git code," 2015. [Online]. Available: https://github.com/tzutalin/labelImg

[19] R. Padilla, W. L. Passos, T. L. B. Dias, S. L. Netto, and E. A. B. da Silva, "A comparative analysis of object detection metrics with a companion open-source toolkit," *Electronics*, vol. 10, no. 3, 2021. [Online]. Available: https://www.mdpi.com/2079-9292/10/3/279