VideoStickers: A Tool for Active Visual Note-taking and Annotation

YINING CAO, HARIHARAN SUBRAMONYAM, EYTAN ADAR

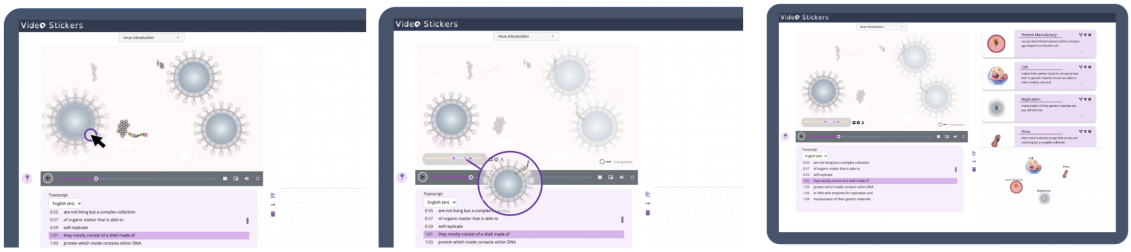

Fig. 1. description

Video is an effective medium for knowledge communication and learning. Unlike linear text in which learners need to connect concepts line-by-line, videos offer an integrated representation of objects and relationships over space and time. However, the high information density in videos makes it challenging to extract specific information during note-taking. Learners need to pause and rewind videos repeatedly, take static screenshots of individual frames, manually annotate motion information, and transcribe and paraphrase during note-taking. The effect is that active learning from videos is often difficult. In this work, we propose VideoStickers, a tool for extracting content from videos as 'motion stickers'. VideoStickers implements automated object detection and tracking, linking objects to transcribed narration, and supports expressive queries to generate stickers across space, time, and events of interest. We demonstrate the utility of VideoStickers for various video topics and notetaking needs.

Additional Key Words and Phrases: visual note taking, video object detection, video interaction, education,

1 INTRODUCTION

Well-designed interactive videos are an effective part of the modern instructional toolkit [5]. Effective narratives, visuals [21] and interactive features [22] have a demonstrated ability to enhance learning outcomes. Unlike linear text, in which learners need to mentally connect concepts one line at a time, videos offer an integrated graphic representation of objects and relationships over space and time. For instance, students can readily observe how a solar eclipse occurs when the moon gets in between the earth and the sun. They can watch different transformations throughout the metamorphosis of a butterfly. They can visually see interactions in complex processes such as protein formation. Developers have recognized these benefits and we now have an ecosystem of robust end-user software platforms for creating (e.g., Powtoons, Animaker, and Moovly) and editing (Apple's iMovie and Final Cut, Adobe's Premiere, Wondershare's Filmora).

Unfortunately, while videos can offer a practical way of visually synthesizing complex pieces of information, they are not always amendable to effective active learning strategies. To address this, developers of educational content often include in-video quizzes. These stops can provide an opportunity for reflection by the viewer. More likely, however, is that they are used as basic attention checks. More robust active learning strategies, such as note-taking [7], are difficult to utilize. In a live lecture, an instructor can modify their teaching to student feedback (explicit and implicit), and adjust pacing to allow students to take notes. Viewers *can* take notes from a video. However, this requires using interactive

video features–pausing, rewinding, or random-access–each incurring a cognitive cost. Worse, the use of these during note-taking can disrupt the learner and reduce the benefits of a well designed visual narrative.

To address this, we propose VideoStickers, a tool for supporting note-taking from videos. VideoStickers utilizes object detection and tracking and linking to narratives to allow the viewer to quickly 'collect' a sticker and integrate the content into graphical notes (see Figure **??**).

VideoStickers is explicitly designed to support graphical notes. Such graphical representations have been demonstrated to support dual coding [25] and improve recall [26]. These representations are also more aligned with the representation used in the video. That is, capturing a graphical image (e.g., of a cell in a video of mitosis) in the notes that corresponds to the representation in the video can enhance retention and provide a useful study tool. However, drawing the image by hand may incur similar costs to producing textual notes. Allowing the viewer to capture screenshots is one solution, but it is often the case that only a part of the frame is interesting for notes. Static captures are also problematic as they remove the dynamic movement that reflects useful content (i.e., multiple static images of the cell–one before and one after a split–may not be as informative as the brief animation of the cell splitting). With VideoStickers, we propose an ability to create dynamic 'clipped' images from the video. By detecting and tracking objects, a viewer can capture short, focused video 'stickers' that can be integrated into a dynamic note sheet or re-used in other ways. By further leveraging the narrative text from the videos, VideoStickers also allowed for rapid collection of textual notes.

In this paper, we propose VideoStickers, a tool for extracting expressive content from videos as 'motion stickers'. VideoStickers implements automated object detection and tracking,linking objects to transcribed narration, and supports expressive queries to generate stickers across space, time, and events of interest. We notice that, the application of visual note-taking is not limited to educational video lessons. We demonstrate the utility of VideoStickers for various video topics and notetaking needs.

2 RELATED WORK

2.1 Interactive note-taking and annotation tools

Note-taking is widely used in learning as a tool for offloading cognitive workload and extend learner's understanding[15]. Current note-taking tools are largely focusing on facilitate information assimilation with text. These works aim to reduce the user's cognitive workload by enhanced efficiency[10] and enriched function[30], allowing better interaction[24] or integrated sense making process [16]. There are a few works contribute to the visual note-taking process by making use of graphic representations. TexSketch[24] allows automatically translate words into sketch images in an active diagramming process. InkAnchor[19] provides a digital ink editor for finger drawing and writing to combine informal graphic content with text and support the capture of informal notes. However, these works are all implement static graphics with limited representation power.

Video is the source of plenty of expressive motion graphics. Extracting existing motion graphics from videos provides a directions for effective visual note-taking. However, video watching is considered a passive and one-time process[4],making information extraction and navigation in videos difficult. The viewers have to go through the complete video to understand the context[11].

There are several existing tools that facilitate an interactive learning from videos by encouraging knowledge sharing in a collaborative learning environment. EdPuzzle[17] allows the learners to edit and add content to videos from a wide variety of online sources. The Vialogues[1] encourages active learning by providing a platform for dialogue. To help with the navigation problem, Interactive Shared Education Environment (ISEE)[18] proposed 'Smartlinks'. It

automatically generates hyperlinked timestamps to associated notes with their video contents. None of these existing systems help with note-taking itself. There is a 'Make-a-Map' function in BrainPOP[20], an animation-based on-line learning environment. This concept mapping tool that allow student to diagram over concepts using images, keywords and movie clips. However, not expressive enough. The users need to either capture the whole screen or not. Moreover, only video format are allowed for animated graphics.

2.2 Video object segmentation and tracking

In a video, there are multiple objects that the audiences might be interested in. In order to extract these expressive contents from video streams, object segmentation and tracking (VOST) algorithms are needed. Current VOST problems including Supervised learning, semi-supervised learning and unsupervised learning, etc.[29]. For the supervised learning system, it will first use a detection model for target localization and then use an embedding model for data association[28]. The supervised learning methods requires a large image training dataset. Though it is accurate and can do localization to detect and track objects separately, can not be applied to arbitrary videos. Semi-supervised learning method separates objects from the background given the mask of the first frame. [2] presents a One-Shot Video Object Segmentation based on tranfer learning from ImageNet. Though these algorithms are really accurate in single object segmentation and tracking, it needs approximated 5-6 seconds to fine-tuning a single frame[2], which can be hardly applied to any real-time system. Unsupervised learning methods do not require user interaction to specify an object to segment. They exploit the information in the frame images and then propagate it to the remainder of the frames by using background substraction[23] or point tracking with use long range trajectory motion similarity and perform clustering over the point[4, 9]. However, the draw backs is that they are not able to segment a specific object due to motion confusions between different instances and dynamic backgrounds[29].

Extracting expressive content as 'motion sticker' is not a typical VOST problem. First of all, there are many motion graphics in the scientific educational videos includes abstraction, metaphors and other distorted transformations. These objects are absent from existing dataset. Thus pre-trained network on large-scale real-world image(e.g. ImageNet[6]) for object detection might fail on extraction. Moreover, we are not require the algorithm to track exactly one single object with fixed characteristics across frames. The extracted motion stickers should be able to capture and show the transformation (e.g. the process of DNA chain transforms into a protein) and interactions (e.g. cells merge, atomic collision).

3 USER EXPERIENCE

To demonstrate the key features and the overall user experience of VideoStickers, we describe the process of creating an animated diagram about 'how corona virus affects our immune system'.The system mainly consists of four parts: video panel, caption panel, stickers panel and diagramming panel. The dropdown menu on the top is for selecting videos.

Select a 'Stickerized' video

Our learner, Eric, should first select a video he wanted to learn using the drop-down menu. The videos in the list are all 'stickerized' through a pre-procesing stage, which we will discussed in detail in later section. After 4 seconds loading for a 2 minutes short video, the VideoStickers system is ready for him to interact with.

Watching and Marking Frame of Interest

For an educational video with high information density, learners can hardly understand the video by merely a one-pass watching. Considering the users might go over the video content once again, the system provides a 'pin-mark' button beside the video control bar for annotation the frames of interest. For example, when the video introduces the two

kinds of immune cells that extremely vulnerable, Eric might want to mark the point for later reference. When he click the 'pin-mark' button, an orange dot will shows on the timeline indicating the frame of interest.

Beneath the video display panel, is the panel for interactive captions. When watching the videos, the caption associated with be highlighted in real time. Eric can easily navigate to the frames related to each sentence by simply click the sentence.

Motion Sticker Extraction

When Eric saw some graphic representation he wants to extract out as stickers, he pauses the video and detected objects will be detached from the video interface as static stickers on top of the original objects. The system will detect multiple objects, and when he hovers over by mouse and he will see each detected object pop out. After a sticker is clicked, Eric will enter an edit view of the selected sticker. In the edit view, only the selected sticker is an active component on top of the screen, with a slider and a background toggle on the bottom of the video display, which allows the user to choose whether to include the context information into the motion sticker.

Eric can click to select the graphic representations of both an object or a process:

Extract object: To get graphic representations of the Corona, the Neutrophiles and the Killer T-cell.

Extract process: To get a graphic illustration of the fibrosis of our lung tissue...(graphic illustration) However, for some graphic representations, the background information is preferred to be included. For example, the process of how corona connects to a specific receptor on its victim's membranes and injects its genetic materials. It is better to show the cell membranes. In this case, Eric can turn on the background and generate the sticker with a more comprehensive context.

Targeting to Point

The default length of a motion sticker is 2 seconds. However, this default motion sticker is not covering all frames that illustrate a process. When Eric extract the sticker of the process describing how corona connects to a specific receptor and injects its genetic materials into the cell. There will be two interaction that he wants to capture in this motion sticker: (1) connects the receptor; (2) inject the genetic materials. VideoStickers offers him a slider with the range covering the entire sequence of frames where the corona appears. On the slider, the potential points of interaction are marked with orange circle. In this case, the points where the 'connects' and 'injection' happens are marked out for him. With the slider and references for points of interest, Eric can efficiently choose the most representative start and end timestamps for the sticker.

If Eric is unsatisfied with the object he selected, he can click the 'cross' button besides the slider, the system will then revert to the 'static sticker' view.

Add Stickers to Panel

Once he has finished creating a satisfied motion sticker, he can clicks the 'plus' button besides the slider and add it to the Sticker Panel. The sticker will be contained in a Sticker Card with automatically generated labels and captions. The labels are the words detected on the screen. For example, the 'Corona', 'Neutrophiles' and 'Killer T cells' are labels for Eric correctly. The caption generated are compact sentences within the time range of the motion sticker. These two text fields are open to free edit. Eric can assign more suitable wordings and phrases based on his needs for understanding and memorizing the content.

Diagramming with Easy Navigation

With a list of generated stickers, Eric is going to diagram over the video content to catch the sequential and casual relationships between the processes and objects. By clicking the 'diagram' button on the top right of the Sticker Card, he is able to send the labeled sticker to the diagramming canvas. He is able to add arrows or texts to the canvas to

indicate relationships and add necessary explanations. In the diagramming process, Eric might find some concepts confusing and want to watch the relevant video content retrospectively. He has two options to do that. He can click the marked dot on the timeline to navigate the certain frames of interest. Or, he can click the 'locate' button on the right top of the Sticker Card to navigate to the start point of the motion sticker.

4 SYSTEM DESCRIPTION

VideoStickers is a tool we developed to facilitate visual note taking process by automatically detecting, tracking and detaching dynamic visual representative objects from the video stream.

4.1 Overview

As a notetaking tool for video watching, our system tackle with several pain points in user's learning experience from videos and proposed specific functions:

(1) **Concept Capturing:** Encode video content into separate lightweight 'motion stickers' with various dimensions including content(region, contour, transformations), spatial arrangement(motion path, potential interaction) and timing of animation(duration,speed).
(2) **Text Association:** Suggest labels and relate transcriptions to specific stickers.
(3) **Interest Point Detection:** Suggest start and end points for the appearance of certain objects and potential points for interest (e.g. interaction point between objects, transformation point)
(4) **Annotation:** Allow users to mark the timeline with Frame of Interest and customize text on top of the stickers.
(5) **Navigation:** Easily trace back to video context with certain stickers.
(6) **Diagramming:** Provide an interface for diagramming over stickers.

4.2 System workflow

This section introduces the overall system workflow for VideoStickers.

4.2.1 Set up stage: In order to guarantee a real-time user interaction. VideoStickers went through a pre-processing stage for each input video. In the pre-processing stage, two tasks need to be done

(1) 'Stickerize' over video: create static stickers for sampled frames with timing, positioning information and save to .json file. For each individual frame, multiple stickers are coded timestamp and index counting from left to right of the display.
(2) Label detection: detect text on sampled frame images and save in .json file

After pre-processing, it takes several seconds for the server to read in all the processed information and be ready for serving the system.

4.2.2 User interaction stage: After the system set up, the user can use interact with a 'stickerized' video in the ways described in detail in the User Experience part. Here, we will illustrate how these interactions are supported by VideoStickers.

Pause to show stickers: When user pause, the current timestemp will be send to the backend to query for the static stickers at current frame. Static stickers corresponding to that timestemp will be displayed on top of the screen, located by the positioning information.

Select sticker: When selection is made, user enter the sticker edit view. Both the index and timestemp of the selected

sticker will be send back to server. The sticker will be tracked over frame until the start and end point of the sticker are detected. The tracking will be no longer than 10s for both forward and backward pass. Within the tracked range, interest points are detected. The the list of interest points will be visualized as dot marks on the slider with the range determined by the start and end point. By default, a 2 seconds motion sticker, starting from current timestemp, will be generated and displayed on top of the video display.

Edit sticker: When the user is not satisfied with the default motion sticker, he/she can change the time range, add text or turn on/off background by the tools provided in the edit view. Coresponding changes will be send to server for updating the sticker. If user de-select the sticker, the system will revert to the multiple sticker selection view.

Add sticker to panel: When the motion sticker is finalized in the edit mode, it can be added to the Sticker Panel with a '+' button besides the slider. The selected time range will be send to server to query for coresponding labels and caption in the time range. There might be multiple labels detected, while the first detection will be used as a suggested label. The suggested label and concatenated transcriptions will be shown in 'Sticker Card' with the 'motion sticker'.

Understanding and diagramming: With timestamp encoded in the motion stickers, users can easily navigate back to the start point of the sticker with a 'locate' button in Sticker Card. Users can do instant diagramming with an embedded canvas on the buttom right of the display. The motion sticker corresponding with the label will be added to the canvas via 'diagram' button.

4.3 Dynamic Visual Representative Object Generation

In this section, we will discuss in detail about how the VideoSticker system detects visual representative objects, tracks it over frames and generates dynamic stickers for arbitrary input videos.

We generate discrete representative objects in each frame with the schema depicted in Figure **??**. Frame images are sampled from the video stream with a sample rate of 0.1. Each image will be sent to Poolnet to derive a saliency map. Then the image will be down sampled at a 0.1 sampling rate. Objects with spatial differences will be segmented with DBSCAN algorithms.

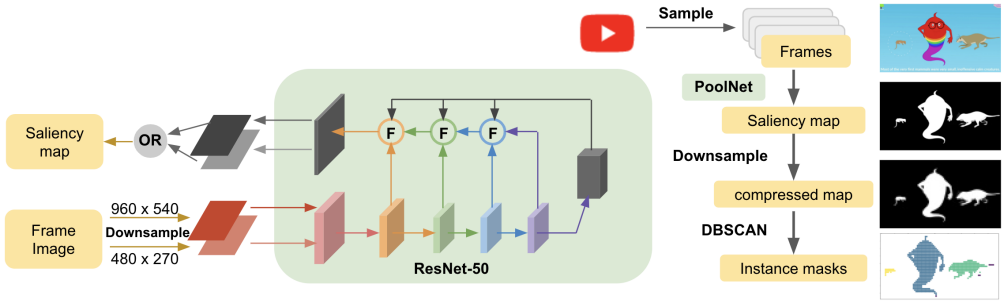

Fig. 2. Dynamic Visual Representative Object Generation Schema

4.3.1 Saliency map generation and refinement. Saliency maps are generated with deep learning techniques. We adapted the network structure of PoolNet[14] to generate binary saliency maps. The PoolNet is based on the U-shape architecture with a global guidance module (GGM) and feature aggregation module (FAM) to achieve a detailed yielding pooling. This network has state-of-art results on salient object detection. The original PoolNet discussed several backbone

choices and training options. The network we integrated into our system is jointly trained with edge detection, which is reported to perform better on ambiguous scenes with low contrast between foreground and background. We choose ResNet-50[8] as backbone and the DUTS dataset [27] as a training set.

However, in implementing the network, we found that the granularity of detected salient objects largely depends on original image resolution. We conducted an experiment over several graphic videos and the DAVIS-2017 [3] dataset, and found the pattern between image resolution and granularity of saliency map: images with lower resolution will result in a general capture of salient objects, while higher resolution images will result in finer grained highlights.

In our system, we are looking at the comprehensiveness, compactness and completeness of the stickers. From the observation of the qualitative experiment results, we choose the resolutions of 960x540p and 480x270p for salient object detection based on the following considerations: (1) 960x540p salient map is more comprehensive on instance level (i.e., as many visual salient objects will be detected); (2) 480x270p salient maps generate more compact and complete masks over objects. We integrate two output saliency maps by a pixel wise 'OR' operation.

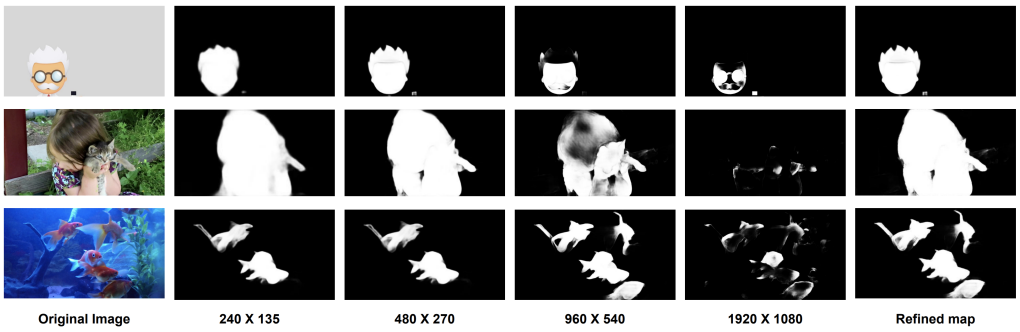

Fig. 3. Qualitative comparisons between image resolution and detected object

4.3.2 Object tracking and interest points detection. We use the median flow[12] algorithm for both forward and backward object tracking. We then compare the bounding box proposed by the tracker $BdBox_t$ and the detected bounding boxes in the pre-processing stage $BdBox_{pre}$ for the sequential frames. Decision is made by the Intersection Over Union(IOU) metrics. The $BdBox_{pre}$ with the largest IOU with the $BdBox_t$ will be chosen. We restrict both the forward and afterward tracking to be shorter than 10 seconds and the forward tracking to be longer than 3 seconds. If the object lost track within 3 seconds, the next frame object will be chosen using IOU with the object in the previous frame. This guarantee a reasonable system response time and compensate for tracking errors (i.e., fail to track the same object or incorrectly track the wrong object).

Interest points are detected within the period. The point of interest(POI) is detected with the following 2 criteria:(1) IOU of the selected detected boundingbox and the one proposed by tracker drops by a threshold(we choose 0.4), this indicates a sudden dis-match of detected object and tracked object, possibly due to merge or transformation. (2) Number of stickers decreases at this frame: this condition indicates a between subjects interaction (e.g. collision).

4.3.3 Facilitate understanding with text. Captions and text labels are always useful for understanding video content. When taking notes based on video, users tend to use the phrases described in the captions and use the label annotated in the video as the name of the representative graph (Assumption of user study results). Thus we integrate the text features into the VideoStickers system in respect to both on frame labels and related captions.

Suggested labels: As visual representations, each dynamic sticker generated in the generation phase. The labels are detected from frame images, and associated with each sticker based on timestamps. We implemented the Optical Character Recognition Algorithm from Google Vision API in the VS system to preprocess labels into a json file.

Caption: Start and end point of the sticker is marked, and captions are associated with stickers aligned with timestamps.

5 RESULTS

5.1 Application Scenarios

We use VideoStickers to create notes on videos based on three different user cases: (1) Educational videos; (2) Instructional videos; (3) Entertainment videos. We show that VideoStickers can capture expressive content from multiple kinds of videos including motion graphic videos, 3D graphic videos, real-world scenarios and cartoons.

Educational Video

For educational movies, learners will first generated sticker notes with labels and explained captions. Then diagram over the motion graphics to create a comprehensive diagram that connects the concepts shown in the stickers. In the 'user experience' session, we showed an example of note-taking process for an video illustrating 'how corona virus affects human's immune system'. Here is another example for neutron stars. For this 103 seconds video clip, we

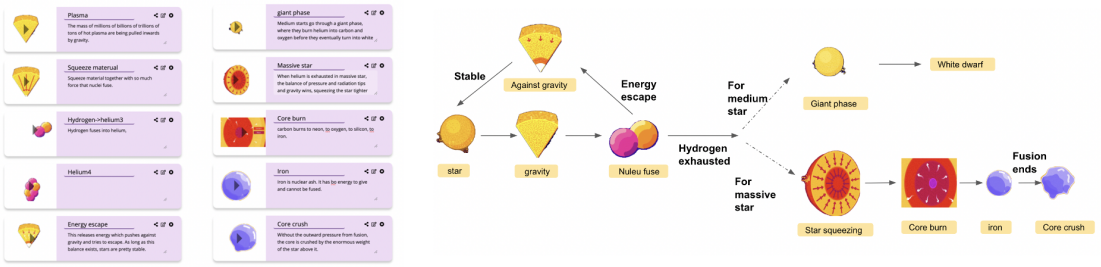

Fig. 4. User Scenario 1

first pre-process it to stickers' the frames, this process takes 35 seconds to finish. Then we load the video and start interaction. As there are many concepts, objects and processes described in the video, we first make sticker notes about each important content. VideoStickers will generate related labels and captions for you in the sticker cards. After we have all the stickers, we start diagramming. When we struggled with some concepts, we can navigate to certain part of the video with related stickers.

Instructional videos

Besides the educational video, VideoStickers can also be applied to record step-by-step videos like recipes or workout instructions using the same process as illustrated in the above session. The generated diagrams and motion stickers can be easily used in blogs or posts for the purpose of knowledge sharing.

Creating Your Own Memes

Memes are widely used in online social communities. VideoStickers can also be applied to entertainment videos for creating memes with a few clicks. In the editing mode, there is an 'A' button besides the slider, which allow users to customize some texts on top of the sticker. The 'A' button will changes to a 'trash' button and a 'download' button when editing the texts. The users can remove the text or download the sticker with text as a meme.

Fig. 5. User Scenario 2 and 3

5.2 Dimensions of Stickers

As graphic representations of certain concepts, VideoStickers creates stickers encoding three different dimensions to facilitate content understanding:

Content

The stickers are generated based on exact masks, rather than fixed bounding boxes. This allow the stickers to capture both the region and contour information of each objects. The contour information is useful to show transformations of objects. For example, the protein folding process shown in (fig.), it vividly shows how a polypeptide chain folds to become a biologically active protein in its native 3D structure. The region information gives intuition of the relative size of objects by comparison. For example, a video introducing the different destine of stars when helium exhausted starts with comparison of medium stars like our sun and massive star. The extracted stickers provide learners with the information of how sun and massive star differ in size.

Spatial Arrangement/ Positioning

Spatial information for each sticker is stored. From the bounding boxes, we can easily calculate the position in videos frame, relative position between objects, motion path and potential interactions.

Timing of Animation

The timing information is encoded in the name of the .gif file of the stickers. Thus, we can easily extract the duration of animation and order of animation. This dimension can be especially useful for measuring speed. As shown in the notes for workout exercise(fig.). It is important to know the speed and intensity of each movement.

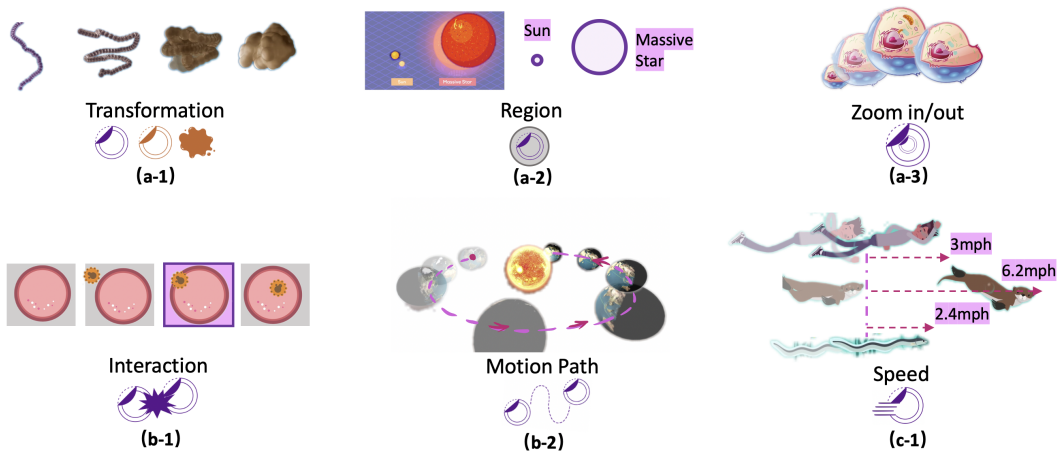

Fig. 6. Dimensions of the video stickers

6 EVALUATION

6.1 Evaluating Sticker Generation

In VideoStickers, Motion Stickers are generated based on the saliency and spatial differences. We combined results from low-resolution images and high-resolution images to capture both the general level and detail level information. This allows us to capture a comprehensive set of objects that might capture the interest of the learners and make the stickers compact and nice-looking.

This algorithm works on different types of videos including motion graphic videos and real-life videos as shown in Figure **??**

The DBSCAN based image clustering algorithm for segmenting objects. Though not optimal as an image segmentation solution, it is suitable for the VideoStickers system. We aim to capture the most expressive components in the frame, rather than exactly the object with certain shapes. Thus when objects interact or collide with each other, it is reasonable to contain them in a single sticker. For example, instead of capturing a single nuclei, we are more interested in its fusion process.

However, there are some failure cases where the frame contains too many discrete objects with frequent change over spaces to demonstrate a certain concept, which will lead to segmentation error. An example including the (transcription process). Some occlusion will cause an incomplete detection of the foreground salient object(example-a recipe video). We compensate inaccurate detection of objects and segmentation by enabling users to turn-on the background and select multiple objects into one sticker to obtain a complete view of the scene to compensate for the inaccurate detection of objects.

6.2 Evaluating Tracking and Interest Point Detection

We track objects with median flow algorithm. As evaluated by an existing study[13], the median flow tracker performs real-time tracking speed with good performance on sequential frame tracking and is good at reporting failures. The tracker is not used for localization. Sequence of object masks are obtained by comparing the proposed bounding

box with detected objects in each frame. We increase the algorithm robustness by limiting the tracking time period. The hybrid method allows us to capture the transformations. For example, the protein folding process (shown in fig). Traditional object tracking algorithms will fail to consider the first frame amino acid chain and the last frame 3D protein as the same object.

The interest points are detected within the range of the target object appearance. By the two criteria proposed in 4.3.2, we are able to detect points of transformations and interactions. Failure cases including False-Positive(FP) cases and False-Negative(FN) cases. FP cases usually occurs due to detection error. Similar to the sticker generation process, when the frame contains too many discrete objects with frequent change over spaces, number of detected stickers will change frequently and trigger multiple times. FN cases are usually caused by a smooth transition. For example, the giant phase of medium star turn into white dwarf. Only the last transformation from red giant to white dwarf will be considered 'interesting', since the previous phases transform so smoothly and might be considered as a zooming process to the system.

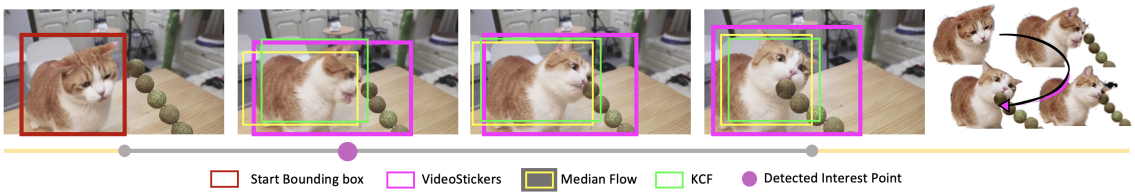

Fig. 7. comparison of different tracking method

7 DISCUSSION

7.1 Towards video understanding and summarization

Contribute to a large scale dataset for video understanding, summarization and keyframe detection.

7.2 From education to social interaction

Animated stickers on media

7.3 Limitations Future work

Enable backend edit by instructors More sticker manipulate operations Refine text integration

8 CONCLUSION

REFERENCES

[1] Megha Agarwala, I-Han Hsiao, Hui Soo Chae, and Gary Natriello. 2012. Vialogues: Videos and dialogues based social learning environment. In *2012 IEEE 12th International Conference on Advanced Learning Technologies*. IEEE, 629–633.

[2] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. 2017. One-shot video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 221–230.

[3] Sergi Caelles, Jordi Pont-Tuset, Federico Perazzi, Alberto Montes, Kevis-Kokitsi Maninis, and Luc Van Gool. 2019. The 2019 DAVIS Challenge on VOS: Unsupervised Multi-Object Segmentation. *arXiv:1905.00737* (2019).

[4] Lin Chen, Jianbing Shen, Wenguan Wang, and Bingbing Ni. 2015. Video object segmentation via dense trajectories. *IEEE Transactions on Multimedia* 17, 12 (2015), 2225–2234.

[5] Ruth C Clark and Chopeta Lyons. 2010. *Graphics for learning: Proven guidelines for planning, designing, and evaluating visuals in training materials*. John Wiley & Sons.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

[7] Deborah DeZure, Matthew Kaplan, and Martha A Deerman. 2001. Research on student notetaking: Implications for faculty and graduate student instructors. *CRLT Occasional Papers* 16 (2001), 1–7.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep residual learning for image recognition. arXiv 2015. *arXiv preprint arXiv:1512.03385* (2015).

[9] David Held, Devin Guillory, Brice Rebsamen, Sebastian Thrun, and Silvio Savarese. 2016. A Probabilistic Framework for Real-time 3D Segmentation using Spatial, Temporal, and Semantic Cues.. In *Robotics: Science and Systems*.

[10] Ken Hinckley, Shengdong Zhao, Raman Sarin, Patrick Baudisch, Edward Cutrell, Michael Shilman, and Desney Tan. 2007. InkSeine: In Situ search for active note taking. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 251–260.

[11] Shruti Jadon and Mahmood Jasim. 2019. Video summarization using keyframe extraction and video skimming. *arXiv preprint arXiv:1910.04792* (2019).

[12] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. 2010. Forward-backward error: Automatic detection of tracking failures. In *2010 20th International Conference on Pattern Recognition*. IEEE, 2756–2759.

[13] Ville Lehtola, Heikki Huttunen, Francois Christophe, and Tommi Mikkonen. 2017. Evaluation of visual tracking algorithms for embedded devices. In *Scandinavian Conference on Image Analysis*. Springer, 88–97.

[14] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang. 2019. A Simple Pooling-Based Design for Real-Time Salient Object Detection. In *IEEE CVPR*.

[15] Tamas Makany, Jonathan Kemp, and Itiel E Dror. 2009. Optimising the use of note-taking as an external cognitive aid for increasing learning. *British Journal of Educational Technology* 40, 4 (2009), 619–635.

[16] Xiaojun Meng, Shengdong Zhao, and Darren Edge. 2016. HyNote: Integrated Concept Mapping and Notetaking. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*. 236–239.

[17] Leann J Mischel. 2019. Watch and learn? Using EDpuzzle to enhance the use of online videos. *Management Teaching Review* 4, 3 (2019), 283–289.

[18] Xiangming Mu. 2010. Towards effective video annotation: An approach to automatically link notes with video content. *Computers & Education* 55, 4 (2010), 1752–1763.

[19] Yi Ren, Yang Li, and Edward Lank. 2014. InkAnchor: enhancing informal ink-based note taking on touchscreen mobile phones. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1123–1132.

[20] Yigal Rosen. 2009. The effects of an animation-based on-line learning environment on transfer of knowledge and on motivation for science and technology learning. *Journal of Educational Computing Research* 40, 4 (2009), 451–467.

[21] Katharina Scheiter, Peter Gerjets, Thomas Huk, Birgit Imhof, and Yvonne Kammerer. 2009. The effects of realism in learning with dynamic visualizations. *Learning and Instruction* 19, 6 (2009), 481–494.

[22] Stephan Schwan and Roland Riempp. 2004. The cognitive benefits of interactive videos: learning to tie nautical knots. *Learning and instruction* 14, 3 (2004), 293–305.

[23] Jianbo Shi and Jitendra Malik. 1998. Motion segmentation and tracking using normalized cuts. In *Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271)*. IEEE, 1154–1160.

[24] Hariharan Subramonyam, Colleen Seifert, Priti Shah, and Eytan Adar. 2020. texSketch: Active Diagramming through Pen-and-Ink Annotations. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13.

[25] N Thomas. 2014. Dual coding and common coding theories of memory. *Stanford Encyclopedia of Philosophy. Retrieved from https://plato. stanford. edu/entries/mental-imagery/theories-memory. html on November* 7 (2014), 2017.

[26] Jeffrey D Wammes, Melissa E Meade, and Myra A Fernandes. 2016. The drawing effect: Evidence for reliable and robust memory benefits in free recall. *The Quarterly Journal of Experimental Psychology* 69, 9 (2016), 1752–1776.

[27] Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. 2017. Learning to detect salient objects with image-level supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 136–145.

[28] Zhongdao Wang, Liang Zheng, Yixuan Liu, and Shengjin Wang. 2019. Towards real-time multi-object tracking. *arXiv preprint arXiv:1909.12605* (2019).

[29] Rui Yao, Guosheng Lin, Shixiong Xia, Jiaqi Zhao, and Yong Zhou. 2019. Video object segmentation and tracking: A survey. *arXiv preprint arXiv:1904.09172* (2019).

[30] Dongwook Yoon, Nicholas Chen, and François Guimbretière. 2013. TextTearing: opening white space for digital ink annotation. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. 107–112.