Review

AI Video Editing: a Survey

Xinrong Zhang [1], Yanghao Li[2,†], Yuxing Han[3,†], Jiangtao Wen [4,†]

1 Tsinghua University; zxr19@mails.tsinghua.edu.cn
2 Tsinghua University; lyh18@mails.tsinghua.edu.cn
3 Tsinghua Shenzhen International Graduate School; yuxinghan@tsinghua-sz.org
4 Tsinghua University; jt-wen@mail.tsinghua.edu.cn
* Correspondence: jt-wen@mail.tsinghua.edu.cn
† These authors contributed equally to this work.

Abstract: Video editing is a high-required job, for it requires skilled artists or workers equipped with plentiful physical strength and multidisciplinary knowledge, such as cinematography, aesthetics. Thus gradually, more and more researches focus on proposing semi-automatical and even fully automatical solutions to reduce workloads. Since those conventional methods are usually designed to follow some simple guidelines, they lack flexibility and capability to learn complex ones. Fortunately, the advances of computer vision and machine learning make up the shortages of traditional approaches and make AI editing feasible. There is no survey to conclude those emerging researches yet. This paper summaries the development history of automatic video editing, and especially the applications of AI in partial and full workflows. We emphasizes video editing and discuss related works from multiple aspects: modality, type of input videos, methology, optimization, dataset, and evaluation metric. Besides, we also summarize the progresses in image editing domain, i.e., style transferring, retargeting, and colorization, and seek for the possibility to transfer those techniques to video domain. Finally, we give a brief conclusion about this survey and explore some open problems.

Keywords: AI, deep learning, video editing, image editing

1. Introduction

Video has become a prime format of media in our daily life. We can see video anywhere at anytime, like in elevators and cinema, and on Reddit and Facebook. In the 3rh quarter of 2021, there are 213 million Netflix[1] subscribers and 200 million Amazon Prime[2] subscribers[3], and over 14 million daily active users of TikTok. And [1] predicts that by 2022, video viewing will account for 82% of all internet traffic. In my view, video consuming is highly like to be under-estimated for COVID-19 makes underline gathering risky and especially inconvenient for international travels, which boosts the demand for online information exchange. Video is becoming indispensable.

Consequently there is a surge in video editing out of diverse motivations. Editing raw footages into a watchable video is the basic aim[2][3]. Extracting highlights from videos is a tradeoff between saving costs, e.g. time, storage space and internet traffics, and preserving their important information[4][5][6]. Those videos are typically long and most of their contents are boring, such as surveillance videos[7], egocentric videos[8]. Videos are always generated, and edited for a purpose. However, when those videos are reused on different occasions, it's necessary to re-edit them. For example, films are usually cut of a certain ratio, e.g. 16:9, 4:3. Retargeting is needed when users play them on displays of various ratios and sizes[9][10]. Another motivation to edit video is to help retrieving information in videos, for that tranditional video retrival needs viewers to watch the video, which is very time-wasting. Sivic et al. [11] first retrive infomation from videos in the manner Google does in text, and it summaries the features of obejcts into a 128-vector, and transfroms the approaches in nature language processing (NLP) field to retrival certaion objects. Besides,

Citation: Xinrong, Z.; Yanghao, L.; Yuxing, H.; Jiangtao, W. AI Video Editing: a Survey. *Preprints* **2021**, *1*, 0. https://doi.org/

Received:
Accepted:
Published:

1 https://www.netflix.com/
2 https://www.amazon.com/amazonprime
3 https://www.statista.com/

transforming 360^o videos into 2D normal field of view (NFOV) videos[12][13][14][15], and adding special effects[16][17] account for a quite portion. Given the huge market potential in video editing, and demands for film and video editor is growing continuously[4], automatically editing is very promising.

Automatically video editing is far from an easy task. First, videos often contains dubbing, underscoring, and subtitles, and editing involves molti-modality, which poses a great challenge. The techniques of a modal may be unable to function well in practical. For example, automatic speech recognition techniques cannot meet the expectation of professional editors. They ususally require a perfect transcript from video providers or crowdsource[18][19][20]. The corelations among multi-modality is not well investigated, though some researches have attempted[21]. Second, it's hard to foresee what audiences want to see. Previous researches has revealed that there are remarkable differences on saliency between viewers while watching the same images[22][23][24][25], and disparities also exist while watching at VR and desktop for the same viewer[26]. Third, video editing should obey the principles of logics and aesthetics that still could not be programmed in computer languages. Socrates concludes that

Beauty is difficult to define.

Even though, researchers have achieved great progresses towards automatically video editing with the help of a lot of efforts, as well as the advances in artificial intelligence(AI) including machine learning, deep learning, computer vision and so on. For example, in audio domain, Rubin et al. [27] propose a semi-automatic audio editing tool, UnderScore, that takes a sequence of speech whose emphasis points are annotated by the user and a music track as inputs. UnderScore then refines the emphasis points according to RMS energy, selects a suitable change point in the music automatically, aligns music with the speech with adjustable length of pre-solo, solo and post-solo and adjusts the dynamics of the composition. Later, they build an audio story editing tool to edits speech and add appropriate music to the speech[28] . It detects breaths, abstracts music features, and predicts music sentiment. Besides, fading in or out is applied. Rubin further investigates the method to generate musical scores for audio stories[29].

In this paper, we will review those efforts and researches that editing existing videos and images rather than rendering or synthesizing from scratch. Besides, we mainly introduce methodology and algorithm and do not compare their performance. First, recent works in each modality: audio, text, and vision, and the combination of modalities. Following two sections to introduce input video domains: lecture, sports, 360^o, web videos and so on, and manipulations: shot-level, frame-level, and object-level. Next impressive datasets and commonly used evaluation metrics. Then AI image editing techniques that could spark amazing ideas on video editing in Section 5. At last, a conclusion and some insights about future researches.

2. Modality

A video might contain multi-modality information, e.g. subtitles, audio tracks, except for frames. Aizawa et al. [30] even collect brain waves of human that wear a camera. In practice, single modality[31][28], bi-modalality[32] or multi-modality methods[33][34] to edit videos are all explored widely. Here, we focus on three common modalities: audio, vision and text.

2.1. Audio

As audio technique starts and grows earlier than that of vision, audio feature is a signficant cue in eariest video editing. Common audio features include pectrograms and spectral flux, onset envelopes, tempo and tempograms and beats [17]. Onset envelopes, also sometimes called novelty curves, are an approximate measure of how likely an onset has

[4] https://www.careerexplorer.com/careers/film-and-video-editor/job-market/

 doi:10.20944/preprints202201.0016.v1

occurred at each point in time. In 2000, Rui et al. [31] extract highlights for TV basketball programs only using audio cues, as well as Xiong et al. [35].

Even when more features are used, audio is still an important factor. For example, Chih et al. [36] utilize audio and motion information. Li et al. [37] use aural-visual cues to analyse football video based on deterministic reasoning and probabilistic inference. Heck et al. [38] use visual and audio features. SceneSkim[19] and QuickCut[39] use audio and text. In [34], multi-modality features are extracted.

Dubbing is a routine of video editing. For example, Leake et al. [40] use Google Cloud Text-to-Speech service, Google Speech-to-Text and Needleman-Wunsch algorithm to generate audio narration. Some works like [41] utilizes augmented Tacotron 2 model for TTS and utilizing a cross-lingual voice clone technical to learn pronunciation and intonation. Casting Words.com and Amzon's Mechanical Turk is also a popular speech-to-text website. A fully automatic method with sentiment analysis techniques is proposed in [29]. Shin et al. [42] propose Voice Script that edits script and audio recordings in a unified way.

Except dubbing, musical score is also a part of audio. Rubin et al. [28] propose an audio story editing tool that navigates and edits speech, selects appropriate music for the score and edits the music to complemnet the speech. Breaths and multiple takes of sentence can be detected. The features of music, like tempo, mode, danceability, timbre are calculated, and its valence/arousal values are predicted with its MFCCs using machine learning. Music is edited based on beats, and can be shortened or lengthened as need. Besides, fading in or out is applied to suit audio story. He also develops P2FA to align transcript with audio. Rubin further proposes a system and algorithm for re-sequencing music tracks to generate emotionally relevant music scores for audio stories[29]. The first stage is to decide the emotions of segments of speech and music. As emotion can be quantified with valence and arousal. Speech is transcripted into text transcript and is segmented by its paragraph boundaries. Then the text is time aligned with speech by a variant of the Penn Phonetics Lab Forced Aligner. For automatic labeling, look up emotion table to get the valence and arousal rating of every word, normalize them by global valence/arousal mean and standard deviation, and compute the average scores of all words in a paragraph to get the emotion label of each paragraph. And the music is segmented by computing a hierarchical clustering of self-similarities in a track and finding an optimal pruning of the cluster tree. The way to label music is same as speech, except that for automatic labeling, a multiple linear regression model. After labeling speech and music, it should design score generation algorithm to generate musical scores. There are two primary costs. Matching costs suggest how well the music emotions match the speech emotions and transition coasts beat-to-beat transitions in the music. The algorithm is aimed to search the lowest cost musical score using dynamic programming. And Ellis' beat tracking algorithm[43] is used to detect the beats in a piece of music. Two 2D tables are constructed. One is speech and music beats match costs in terms of L2 distance between emotions of speech and music and the other is the transition cost from one beat to another beatin terms of timbre, pitch and volume distances. Besides, some structral constraints like pauses and limiting music segment lengths are also considered, as well as stylistic constraints like minimum loop constraint, musical underlays and multiple music tracks. The values of entries of the two tables is decided by authors' judgements. The cost function is the sum of match costs and transition costs. Once the minimum cost solution is found, music with some preprocessing like fading in or out and speech emphasis is generated.

2.2. Vision

As the computer vision advances, visual features become dominant. More and more researches take advantages of visual features to promote their works. Lu et al. [8] only make use of visual features. Shin et al. [44] use text and visual features, as well as Wang et al. [32].

The main visual features can be divided into two classes: low-level and high-level. Low-level features are classical and consist of coarseness, contrast, directionality, line-

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

likeness, regularity, color histogram, edge histogram, roughness, and so on[45]. A number of tranditional algorithms are about classical feature extraction: MPEG-7 color layout descriptor[46], CENTRIST[47], VLFeat[48], SIFT[49] and HSV color moments[50]. Gist[51] emphises on learning spatial structures, and works well in descripting natural scene. Histograms of Oriented Gradient (HOG)[52] contains 5 steps: gamma or colour normalization, gradient computation, spatial or orientation binning, local contrast normalization and descriptor block partition.

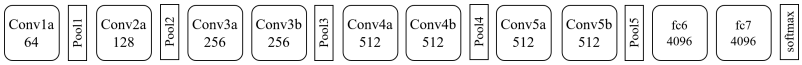

Figure 1. C3D network sarchitecture [53].

High-level features are machine-learning based, and feature extractors generate high-level features directly from input image and video, like C3D[53], or from low-level features, like Viola-Jones face detector[54]. Feature detectors can be divided by their outputs. Saliency maps could be generated by CNN-based models[23][55][56][13], and LSTM-based models[57][58][59][24]. [26][60][61] explore 360^o video saliency. Facial features are detected by VGG-Face model[62], Viola-Jones face detector[54], and other models[63]. Human pose detectors include [64], [65], and Poselets[66]. Object detection can realized with Detect-and-track[67], Faster R-CNN[68], VGG model[69], GoogLeNet model[70], BVLC CaffeNet[71]. Jiang et al. [72] give a comprehensive study on bag-of-visual-words. Some works analyze the sentiment of videos[73].

Fartash et al. [21] propose VSE++, a visual-sementic embedding that provides convenience across modality. Let $\phi(i;\theta_\phi) \in R^{D_\phi}$ denotes the feature of image i, and $\psi(c;\theta_\psi) \in R^{D_\psi}$ represents the feature of text c, and let the joint embedding space be linear projections:

$$f(i;W_f,\theta_\phi) = W_f^T\phi(i;\theta_\phi) \tag{1}$$

$$g(c;W_g,\theta_\psi) = W_g^T\psi(c;\theta_\psi) \tag{2}$$

where $W_f^T \in R^{D_\phi \times D}$ and $W_g^T \in R^{D_\psi \times D}$. And the similarity between image i and text c is the inner product:

$$s(i,c) = f(i;W_f,\theta_\phi) \cdot g(c;W_g,\theta_\psi) \tag{3}$$

2.3. Text

Text acts as a main or support modality in video editing. For example, Pavel et al. [74] propose a semi-automatic system for abstracting digests of informational lecture videos using transcript-based interactions. And he proposes SceneSkim to summarize videos based on their lines.Transcripts is provided or obtained by rev.com[5]. Leake et al. [20] take a perfect transcript and multiple video recordings of a dialogue-driven scene as inputs. It segments recordings into pieces for each line of dialogue. [18] obtains transcripts from audio using rev.com and spots locations and visually significant entities(VSEs) using Google NLP toolkit[6]. [75] uses titles as import cues to summarize videos.

There are also many works ultilizing multi-modality. Wang et al. [32] propose text-driven video editing algorithm that finds keywords in input text. It uses visual-semantic embedding with VSE++[21] to encode text and image and then calculate the similarity between those two modalities. [76] is able to capture the relations between words in corpus text. And Alayrac et al. [77] use it to extract main steps in instruction videos. Xu et al. [78] propose topic extraction model based on non-negative matrix factorization. Sener et al.

[5] https://www.rev.com/
[6] https://cloud.google.com/natural-language/

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

[33] aim at understanding videos using visual and language hints to parse a video into semantic steps with textual description for each one in an unsupervised way. Malmaud et al. [79] perform an alignmnet between textual instruction and instructional cooking videos with text and speech and create two corpuses, one of aligned recipe-video pairs and the other short video clips with a cooking action and a noun phrase labeled. WaveNet[80], a stacked dilated causal convolutional network, transforms text to speech and generate music. Truong et al. [81] combine transcript, the sound source analysis and mouth motion to identify the speaking face[82]. Cour uses dialog information to assign names to faces on the screen[83]. Similar work is [84] that uses subtitle and script text as weak supervision.

3. Input Video Domain

The typical workflow of editing a video consists of clip segmentation, feature extraction and tagging, clustering by themas and selection according to some criteria. Debudding need to keep pace with the scene and undersocre should be appropriate. While editing multiple input tracks, cinematography is an important factor. Scene changes must obey some rules, otherwise the outputs will cause audiences dizzy. Those rules include 180^o rule. Some researches have already reveal them. For example, Serrano et al. [25] conduct user expriments on movie editing continuity in space, time and action.

Above is the general workflow of video editing. For specific type of videos, editing solutions should adapt for it has fixed patterns. Truong points that make-up videos are always focus on face, and the parts manipulated are from the whole face to eyes, lips and eyebrows[34]. Besides, the make-up productions appear at certain steps. Similarly, Chi concludes that instructional videos for physical tasks are from a mess to a production, and the unused components are reducing over the procedure[85]. Thus, many editing solutions are designed for a certain genre. Here, we will introduce editing algorithms according to their input kinds.

3.1. Instructional Videos

Malmaud et al. [86] propose a discrete-time, partially observed, object-oriented Markov Decision Process aimed to reveal latent context in a cooking recipe in text, image or video that always elids many keypoints like action or state transition. This proposal models three conditional probabilities: 1) the probability of next state of objects given temporal action of a semantic frame and state; 2) the probability of action with temporal states and text sentences based on semantic role labeling; 3) the probability of action shown in a video clip. In 2015, Malmaud extends his previous work, and performs an alignmnet between textual instruction and instructional cooking videos with text and speech and create two corpuses, one of aligned recipe-video pairs and the other short video clips with a cooking action and a noun phrase labeled[79]. He pre-processes videos and its user-uploaded textual recipe using a Bayes classifier model to get ASR tokens and parse the recipe text with NLP model to solve zero anaphora problem. There are several ways to align: 1) A HMM is trained to align each step of the recipe to a sequence of words in ASR transcript and apply the Viterbi algorithm to estimate the MAP sequence. Finally align extracted recipe to video segments; 2) another approach to labeling video segments is keyword spotting, namely searching for verbs in ASR transcript and finding its corresponding position in the video with a fixed-size sliding windows. As errors in ASR, keyword spotting does not work well. The third way is to perform keyword spotting for the action in the ASR transcript as before, but use the HMM alignment to infer the corresponding object. However, it can still be refined further. So a visual food detector is trained based on deep learning, feed input video clips to it to get probabilities of all candidates coming from methods above and the match object is selected.

DemoCut[85], a semi-automatic instructional video editing tools for physical tasks, provides a light-weight annotation-based interface, adds temporal effects and visual effects as markers. This system processes single take and single camera footage. Users need to offer five types of markers: step, action, closeup, supply and cut-out. Based on those markers,

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

Democut automatically applies effects. Alayrac proposes a novel unsupervised learning approach that combines the features of the input video and the associated narration to discover and locate the main steps in instructional videos[77].

Sener et al. [33] aim at understanding videos using visual and language hints to parse a video into semantic steps with textual description for each one in an unsupervised way. This paper discovers semantic steps from a video category and adopts a multimodal joint vision-language model for video parsing. Visual atoms are clusters by joint extension to spectral clustering of object proposals generated with Constraint Parametric Min-Cut algorithm, whereas language atoms are frequent salient words with tf-idf measure. And each frame is represented by the occurrance of atoms. It ultilizes Markov Chain Monte Carlo to learn and infer the Beta process Hidden Markov models for understanding of the time-series information.

Truong et al. [34] propose a multi-model method to automatically generate two-level hierarchical tutorials from instructional makeup videos, which allows users navigate by click and voice commands. Two levels are coarse events about objects and fine one about actions that manipulate those objects. In this meaning, makeup videos' high level is about facial parts: lips, eyes and face. While in the coarse-level event, each fine-grain action step consists of a sequence of demonstrating and narration. Besides, there are non-instructional introduction and conclusion. The system takes videos and its aligned transcript as inputs, oversegments and labels them, and makes shot-phrase pairs. According to product introduction pair, it constructs action steps. And according to the number of times each facial part appears, construct facial part groupings.

3.2. First-person and Sports Videos

First-person videos are generated from wearable or portable camares that provide great convenience for non-professional users, so that those videos contains motion blurs, obejct occlusion, shake and other factors that degrade video quality. Thus, camera stabilization is a necessary step. Neel et al. [87] smooth camera motion and speed up videos of hand-held cameras jointly. It first evaluates the matching degree of each frame with its agjacent frames based on sparse feature macth[88][89], select the optimal path with a dynamic-time-warping algorithm, and smooth the path and render the output video. As Kopf does in [90], scene reconstruction, path planning and image-based rendering are basical steps. Sun aims at generating montages from unconstrained videos that are inconsistent and probablily contains motion blur and shake, like egocentric cameras[91][92]. The output montageable image contains the salient person with his salient actions from multiple frames. The chanllages behind this algorithm are human body detection and tracking, salient person detection, and action composition. Hamza et al. [93] focus on address the videos generated from wireless capsule endoscopy.

Top two characteristics of sports videos are audio and motion. For example, [31][35] only use audio features such as MFCC, energy, to extract highlights of basketball games. While Li et al. [37] try to analyze football video based on deterministic reasoning and probabilistic inference with additional viusal cues. However, those algorithm only identify and capture simple rules, but latent and complex rules are still unknown. Thanks to the advances in computer vision and machine learning, more and more algorithms or networks are designed to learn those rules and bring out the huge improvement in video editing. Hanjalic et al. first propose excitement model for highlights extraction from sports videos[94]. This model ultilizes motion, scene cut frequency, and energy of audio to fits a smooth curve. And the most exciting moments are desired highlights. Sun et al. [95] propose a novel algorithm to score the highlightness of sports video clips.

3.3. Animation and Film

As animation and film are both like to be well edited already, re-editing mainly includes retargeting to fit different ratios, or super-resolution for old movies of poor quality.

 doi:10.20944/preprints202201.0016.v1

Galvane et al. [96] present a detailed formalization of continuity editing for 3D animation, proposes an automatical editing method based on semi-Markov assumption in which parameters can be controlled and validates this method through a user evaluation. Given a 3D animated scene and rushes from M cameras and manually annotated time-aligned actions(subject, verb, object), a semi-Markov chain is built on editing graph with node actions and edge costs that are decided by three guidelines(errors in conveying unfolding actions in each shot, violations of continuity editing rules in each cut and errors in shot durations). Then the editing is an optimization problem that can be solved by dynamic programming. A scene from "Back to the Future' is chosen as inputs, 21 viewers show that editing has an impact on the preceived quality of the observed video stimulus, but the preceived quality of the version done by an expert cinematographer is not significantly higher than the new method.

Pavel et al. [19] develop SceneSkim, a system with UI that generates captions, scripts, and plot summaries of movies to support quickly searching and retrival. It aligns speech audio with the caption text with P2FA[28]. It uses search time and accuracy as evaluation metrics. Jain et al. [97] also crops movies. [98] aslo edits videos to different ratios. Khoenkaw et al. [99] use the film gammer of each shot to guide cinematic feature extraction and generate the importance map at server end. At the client end, a cropper retargets the film to desired size. Shots are classifed according to the camera behavior, and objects in each shot is further classified by their movements.

3.4. Lecture Videos

Lectures are of limited space and fixed process. The speaker is standing at platform, back to a blackboard, towards to a group of audiences, probably playing slides. For broadcasting, remote seminar and so on, many researchers devote themselves to proposing a fully automatic plan from recording lecture, editing to post-processing.

Mukhopadhyay et al. [2] come up with *Cornell Lecture Browser*, a automatic system that records lectures and generates multimedia representations. An overview camera records the entire lecture dais, and a tracking camera captures the closeup shot of the speaker. After recording the lecture, editing requires this system solve two key problems: synchronization and automatic editing. The synchoronization between two video tracks is done by adding a synchronization point artificially in one sound channel, the synchronization between slides and videos, and that between the slide titles and the slides is based on feature matching. *Cornell Lecture Browser* defines three principles to constrain the length of shots. Thus, an edit decision list of shots can be calculated by two passes. At last Dalí algorithm[100]selects the final shots from the edit decision list.

Gleicher et al. [101] gives a detailed analysis about challenges and requiments of the framework for virtual videography in lecture domain. And later Heck et al. [38] present an automatical lecture video editing system, Virtual Videography, consisting of four phases: media analysis, attention model, computational cinematographer and image synthesis component. In the media analysis, the input video is segmented into foreground and background based on color to get a clean board stream, region objects on the board are identified, the instructor's gestures are recognized into three types: pointing left, pointing right and reaching up, and audio analysis determines whether the instructor is speaking at a given frame. The attention model determines which regions are important by a few guidelines, and no complex models are used. In the computational cinematographer, a virtual camera determines the type of a shot in the source footage, camera tracking is applied, two kinds of video effects, ghosting shots and picture-in-picture shots are added, best shot sequence is solved by an optimization problem based on a graph and some transitions like fade, pan and zoom are applied to some shots. In the image synthesis, bicubic interpolation is used to obtain high-quality results and other parameters are adjusted to keep coherence. Note that video aspect ratio changes. There are several points to improve. First, the length of the video is not changed, and editing should remove unnecessary clips. What import regions are is decided simply and some novel algorithms can be used.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

Zhang et al. [3] propose iCam2, a fully automatic lecture capture system that supports capturing, broadcasting, viewing, archiving and searching of presentations. It equips with two microphone recording the speaker and audiences, and three cameras for the speaker, audiences and slides and uses five-state finite state machine to change shot among the speaker, audiences and slides.

Pavel[74] proposes a semi-automatic system for abstracting digests of informational lecture videos using transcript-based interactions. The generated digest affords browsing and skimming through a textbook-inspired chapter/section organization of the video content. Input video is segmented into chapters, and the chapter is further segmented in sections. Every section consists of several videos segments with corresponding keyframes and brief text summaries. The text transcript of the input video is supplied by the video or obtained with rev.com. Users select video points or text points to segment videos. Or the system uses BSeg twice to automatically segment text into sections and chapters, as well as videos. However, Summaries are written by human.

Shin et al. [44] present *Visual Transcripts*, a system that transforms a blackboard-style lecture video with transcript into a visual transcript interleaving visual content with corresponding text. Ranjan et al. [102] propose the system that takes the outputs of several cameras and microphones and a motion capture system for meeting as inputs, and generates an edited output video. The system is iteratively refined according to three criteria and advises from experts. The three criteria are 1) it must capture enough visual information; 2) it must be compelling to watch; 3) it must not require substantial human effort. The final prototype design is shown below. In an informal meeting scenario, three participants with a microphone to record audio and a headband to track location and motion sit around a table, with a whiteboard close to the table, and three cameras record the meeting. Four types of shots, close-up shot, two-person shot, overview shot and shot of artifacts are defined and the transitions between three types except shot of artifacts are also fixed. Note that gaze and speaker history are leveraged for prediction. It also restricts camera control.

3.5. Dialogue Videos

Dialogue videos usually contain closeup shot of speakers and is driven by lines. There are two keypoints of dialogue video editing: cinematography for human and text-driven editing. He et al. [103] outline several guidelines explicitly. For example, there are five useful camera distances concerning cutting heights of actors. Cutting at the neck is extreme closeup, under the chest or at the waist closeup, at the crotch or under the knees medium view, the entire person full view and distance perspective long view. But cutting at ankles is very ugly. It also lists some constraints: don't cross the line, avoid jump cuts, use establishing shots, let the actor lead and break movement.

Leake et al. [20] take a perfect transcript and multiple video recordings of a dialogue-driven scene as inputs. It segments recordings into pieces for each line of dialogue, and then concates them as the idioms selected users. There are 13 kinds of basic idioms: avoid jump cuts, change zoom gradually, emphasize character, intensity emotion, mirror position, peaks and valleys, performance fast/slow, performance loud/quiet, short lines, speaker visible, start wide, zoom consistent and zoon in/out. As the concating process is modelled as a Hidden Markov Model, each idiom corresponds to different start probabilities, transition probabilities or emission probabilities. If seversal basic idioms are used, just take an element-wise product of their corresponding HMM parameters start probability, transition probability or emission probability with weights and normalize them. Besides, silence before and after a line can control videos' style and tone.

Berthouzoz et al. [104] present a semi-automatic system with interface to help placing cuts and transition in interview video. Given an interview video, users can obtain its transcription from castingwords.com, and align text to audio with Virage[7]. The system

[7] http://www.virage.com/

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

suggests appropriate cut locations by calculating cut suitability score, and peforms cutting where users delete text. The cut suitability score is:

$$S(i) = S_a(i)S_v(i) \tag{4}$$

where i refers to i_{th} frame, $S_a(i)$ is cut suitability score of audio and $S_v(i)$ is cut suitability score of video. $S_a(i)$ is set to 1 when frame i is between two words, otherwise set to 0. As interview video mainly contains human, $S_v(i)$ is in terms of the distance of mouth, eyes and body between two framescite[52][63]. Once finishes cutting, it generates visible transitions, pauses and hidden transitions, and users decide which one to use. Hidden transition is seamless and needs the system to compute dense optical flow and figure out the number of frames to be interpolated by a data-driven approach. Pauses are generated similarly to hidden transition, and its corresponding audio is copy from the background noise of the environment. While visual transition results in noticeable changes, and consists of jump-cut, fade and so on.

Truong et al. [81] propose ConvCut to generate shareable highlights of 360^o video of social conversations to get rid of headsets. The transcript is obtained from rev.com, and is algined to the audio tracks with phoneme-mapping method. Then ConvCut splits the raw 360^o video into one clip per sentence. With multi-modal analysis, ConvCut gets the information of clips, such as the spatial location of faces[82], the topics of conversation[78], laughter, gestural motion[64][65], facial expression changes. Users select lines and ConvCut automatically edits the corresponding clips into a normal-field-of-view (NFOV) video.

Cheng et al. [105] propose CREATE model to restore the dropped frames of talking video streams whose audio is complete. Given input video with dropped frames and its complete audio, CREATE first aligns frames with audio and figures out which frames are lost using the mounth shape and motion. Based on the rest frames, GAN is used to generate frames that correspond to the audio.

3.6. 360^o Videos

Sitzmann et al. [26] apply saliency prediction algorithm of 2D videos in VR and find that equirectangular projection works better than cube map and patch-base methods and that ML-Net with equator bias predicts better than naive equator bias and SalNet plus equator bias. Besides, it also explores to predict time-dependent saliency with a window and speed, and such method achieves 0.57 average CC score. Based on previous insights and a test with 0.50 CC score, it defines that head orientation can be used for saliency prediction. Forth and last, it explores several simple applications of VR saliency prediction, automatic alignment of cuts in VR video with maximizing the correlation between the saliency maps of the last frame in the first segment and the first frame of the second one, panorame thumbnails, panorama video synopsis and saliency-aware VR image compression.

In [14], Su et al. define the problem that automatically generation of NFOV videos from a 360^o videos as Pano2Vid, and propose AUTOCAM, a system that takes a dynamic panoramic 360^o videos as inputs and outputs several natural-looking NFOV videos. It defines the minimal clip is a glimpse that is of fixed(like 65.5) horizontal angle and fixed aspect ratio(4:3) and is 5 seconds long. AUTOCAM learns the latent essence of capture-worthiness from a great deal of NFOV videos downloaded from Youtube using convolutional 3D, and then predicts the capture worthiness of all candiates of a 360^o video with a classifier. At the first step, AUTOCAM determines the wortiest glimpse. And find the next best glipse at a certain area around the glimpse for avoiding large jump in continuty. This algorithm is weak supervised. [12] extends AUTOCAM in three aspects. First it generalizes the task of Pano2Vid to allow spatial selections within the 360 video and multiple FOVs(104.3, 65.5, 46.4). A 360 video is divided into many ST-glimpses and their capture-worthiness scores are predicted by a logistic regression. Second it presents a coarse-to-fine search approach that iteratively refines the camera control while reducing the search space in each iteration to find a bset trajectory. Third it explores sampling a time window and forbiding

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

the trajectories of the current iteration from selecting the same ST-glimpses as the solution of previous iterations in the window to generate a diverse set of plausible output NFOV videos.

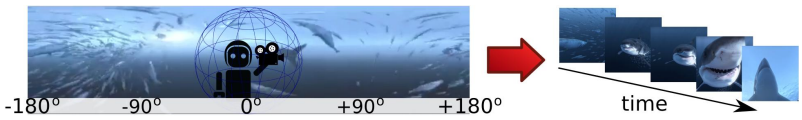

Figure 2. Pano2Vid[14].

Pavel et al. [13] explore two kinds of shot orientation controls for 360^o videos: viewpoint-oriented and active reorientation technique. Given a spherical 360^o video, cut times for each shot boundary and the location in the panorama of one or more important point within each shot, the viewpoint-oriented technique guarantees viewers to initially see the most important content in the shot at each shot change, whereas active reorientation technique allows viewers to reorient the shot by pressing a button so that the important content lies in their field of view. [15] proposes a novel deep learning-based agent for piloting through 360^o sports videos automatically, which leverages a state-of-the-art object detector Faster R-CNN to proposes a few candidate objects of interest, selects the main object with a recurrent neural network, and regresses a shift in viewing angle to move to the next one.

Lai et al. [106] propose a system for converting a 360^o video into a NFOV hyperlapse sampled non-uniformly in space and time using visual saliency and semantics. The input 360 video is first stabilized and the focus of expansion(FOE) is estimated to track the forward camera motion, and next is over-segmented into temporal superpixels whose saliency score is its distance to other TSPs' color and motion. Fully convolutional network (FCN)[107] is used to label temporal superpixels (TSPs), and the top-3 TSPs with maximum scores are detected as ROIs. And users could decide ROIs by UI. After analyzing video contents, camera path planning consists of three phases. Given the detected ROIs and FOE, the smooth camera path is extracted by minimizing a cost function concerning them. In the second phase, 360^o video is rendered into an NFOV video with a fixed field-of-view and then select a set of optimal frames in terms of saliency scores, frame alignment errors, speed and acceleration penalties. When the optimal frames are selected, zooming effect is added according to user perferences and the size of interesting regions. To stabilize 2D NFOV videos, a set of feature trajectories is extraced by the Harris corners and Brief descriptors[89], three motion models for each frame is computed by the RANSAC method[88], AIC[108] selects the best motion models for each frame, and the single-path scheme by the polish the camera motion[109].

Tang et al. [110] come up with a solution to direct 360^o videos containing 5 steps: feature tracking, keyframe selection, motion estimation, keyframe path planning with cinematographic constraints, joint optimization. They proposes a new motion estimation algorithm based on the feature correspondances to handle the rotation and translation between adjacent keyframes. Besides, it also provides a unified framework to define constraints on the outputs through clicking on the ER projection or recording a guided viewing session.

3.7. Surveillance and Multi-view Videos

Surveillance cameras are fixed, and its videos without motion blur or shake. The main problem is to reduce redundancy in surveillance videos. Panda et al. [7] aim at summarizing multi-view surveillance videos without any posits. The multiple surveillance video networks always have an overlapping fields-of-view. It first segments videos into shots according to RGB and HVS color space changes, and extracts their C3D features using [53]. The features of shots are embedded using subspace clustering[111] and then sparse representative selction is performed. Those two steps are performed jointly. Half-quadratic optimization techniques [112][113] are used to solve that optimization problem. Then it

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

gets the optima; sparse coefficient matrix, a weight curve using L2 norms of the rows in the matrix is generated and optimal summary segments are extracted at the local maxima from the curve. It uses 6 multi-view datasets with 36 videos from [6][5].

Truong et al. [39] design a semi-automatic editing tool, QuickCut. It takes a collection of raw video footages and an audio reocrding of the narrated voiceover as inputs, as well as a audio recording of users who speak out the editing actions and objects in the scene while watching the footages. And feed such audio annotations to rev.com or Google's free Web Speech API to obtain corresponding text transcripts and QuickCut time-aligns the text to the raw video using Rubin's algorithm. QuickCut interface provides transcript view pane, footage selctor pane and structured timeline. QuickCut[39] segments video based on motion refering to luminance and then refines segmentation by audio annotations. TF-IDF is used to search for relevant raw footage with etxt queries. Given a set of alignmnet constraints, the problem is to place aesthetically pleasing cuts together to minimize the combined const in terms of frame quality(blurry footage, camera shake and jump cut) and transition. Then use dynamic programming to solve this problem with restrictions. This system is not fully aumatically and mainly based on text and audio rather than video itself. Further, it does not propose a feasible speech-to-text solution so that extra labors are needed.

Heck et al. [38] take several recordings of unattended, stationary video cameras and microphones. Arev et al. [114] present an automatic editing of footage taking multiple social cameras' takes as inputs. The overall cut pipelines are 3D camera pose estimation using a standard structure-from-motion algorithm; 3D gaze concurrences estimation using gaze clustering algorithm of Park[60] to extract 3D points of joint attention(JA-point); trellis graph construction with node camera JA pair and its cost in terms of stabilization, camera roll, joint attention and global vector, edge with weights about transition angle, distance between two cameras, speed of JA-point and size of shot; graph contraction according to user preferences on length, multiple sub-scenes, first person viewpoint and algorithm parameters; path computation with an adptive dynamic programming algorithm to control output video length; and at last rendering. 10 different scenes from 3 datasets using 3 to 18 cameras are edited by this automatic algorithm, a baseline method of cutting vevery three seconds to a randomly chosen camera and a professional movie editor. The conclusion got by authors is that the result vidoes of this automatic algorithm and taht of a professional editor are similar in spirit, although understandably, not identical. But JA-point is not well equl to the most important point. Besides, audio is not used in this algorithm.

3.8. Web Videos

Web videos are from users around the world, and the quality, size, ratio, content of them varies greatly. Investigating their potential value is a hot topic. Sun et al. [95] take advantages of the huge volume video data online, and select edited videos as better to train a ranking model. *Browsing Companion*[115] with UI, collects videos of the same topics online, and discovers the relationship among videos using an HMM model trained with bag-of-words, spatial pyramid and color histogram similarities of frames. So when users watch a video, then could shift to other videos that are related to the current frame. Besides, *Browsing Companion* provides a new solution to abstract highlights from a collection of videos that are unique within videos but common among all videos. Huber et al. [116] propose a transcript-driven B-roll inserting system with a recommendation algorithm for vlogs. First, it analysize popular vlogs online to learn the appropriate locations to insert B-roll with and the relationship between words in transcript and B-roll with SVM . Then it recommand start words and its corresponding B-roll from Giphy[8] and Adobe Stock[9].

Write-A-Video[32], a system with UI that takes user-edited text as inputs, automatically searches for semantically matching candidate shots from input video repository and

[8] https://www.giphy.com
[9] https://stock.adobe.com

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

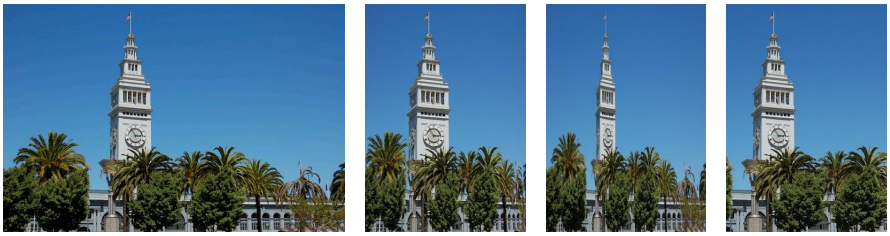

Figure 3. From left to right, original image, its retargeting results with seam carving, scaling and cropping respectively[119].

assembles the video montage with a hybrid optimization objective consisting of shot-wise, cut-wise, and segment-wise energies. For each themed text, keywords used to label the segmented text and to index video shots are given by users. Input videos are segmented into shots using the difference of their histograms in HVS color space. The similarity between text segments and video shots is computed using the visual-semantica embedding with hard negatives(VSE++) approach. It proposes a cinematography-aware assembly algorithm that depends on 2D-based camera motion estimation and tone analysis. And a dynamic programming solver is used to find the optimal shot sequences. QuickCut[39] supports two main modes, alternatives mode and ordered-grouping mode. The alternatives one only assembles one shot for a segmented script, and the ordered-grouping one optimizes cut positions for a manually determined shot sequence. While, the optimization method of Write-A-Video automatically decides the shot sequence order and cut positions. In addition to QiuckCut, Write-A-Video also considers saturation and brightness, oppposite movements and shot duration.

Wang et al. [117] proposes a novel algirthm to edit online short videos driven by paragrah. Given an input paragraph, its sentences are encoded by a bi-directional LSTM. Web videos are represented by their salient objects encoded by LSTMs or NetVLAD[118], so that a matching model between videos and sentences are trained in supervised way. A proposal module recommands top-k matching videos for each sentence, and sorting module arranges the matched videos according to the storyline of the input paragraph using the Sinkhorn network.

4. Editing Manipulation

There are several categorization methods for video editing. Editing ranges from volume-level, frame-level to object-level. Volume-level editing inserts or deletes clips of the input video, selects a set of frames or shots, or remove volume patches. Frame-level editing includes colorization, style transferring and so on. While mainly modifying objects within frames is object-level. It also varies with the input video types as discussed above. Here, we classify editing algorithms according to their manipulation: retargeting that changes the size and ratio of video, summarization that shortens the length of video greatly, and adding special effects that mainly changes the content within frames.

4.1. Video Retargeting

For different displays require various sizes and ratios of the same video, so that retargeting becomes a practical need. The manipulations of retargeting can be divided into 5 classes: cropping, scaling, browsing, seam carving, and warping[120]. Figure 3 shows the differences between retargeting ways.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

4.1.1. Cropping

Figure 4. Retargeting a widescreen recording to smaller aspect ratios. The original recording with overlaid eye gaze data from multiple users (each viewer is a unique color) and the results computed by [98] are shown.

Before cropping, the ROI or saliency map of each frame should be figured out. Once determining the ROI, cropper should smooth results by considering the motion on screen and virtual cinematography. Feng et al. [9] crop each frame and pan it to desired size. He combines motion and extended salience and object identification methods from still images to determine the ROI, but does not consider the temporal dependent among frames over the full video. He defines retargeting loss as the sum of information loss, scaling, pixel aspect ratio, face cut cost, edge crowding cost, pan and cut costs and user hint costs, and uses brute-force search to find optimal solution. Jain et al. [97] optimize the path of a croppping window with three primary operations(pan, cut and zoom) based on the collected eyetracking data. It selects the ROI by learning viewers' gaze data with a RANSAC algorithm[88]. And to evaluate this algorithm, it compares the eye tracking data of original films and the retargeted ones. After optimizing cropping window path, Kiess et al. [121] add some seam carving operations. Rachavarapu et al. [98] retarget videos using eye tracking. Its workflow is as below. First it determines where the new cuts are using gaze data via dynamic programming. It optimizes the cropping window path according to the principles of cinematography, and uses L(1) regularized convex optimization solver. Khoenkaw et al. [99] use the film gammer of each shot to guide cinematic feature extraction and generate the importance map at server end. At the client end, a cropper retargets the film to desired size. Shots are classifed according to the camera behavior, and objects in each shot is further classified by their movements. Li et al. [122] seek for an optimal cropping window path that preserves spatial-temporal saliency and faces with a Max-Flow/Min-Cut method. Liu et al. [123] take more factors to generate saliency map: rate of focused attention, total saliency score, and bias from center penalty.

4.1.2. Scaling

Scaling is keeping the ratio of width to height unchanged while modifying them. Li et al. [124] come up with retargeting videos by segmenting video into spatiotemporal grids. They use grid flow to select keyframes, and resize the grid flows in those keyframes. The left frames are resized by simply interpolating their grid contents from the two nearest retargeted key-frames. This algorithm is of low computational complexity and could preserve the shapes of salient objetcs along time axis. Wang et al. [125] resize each frame of a video independently based on their saliency objects, and then optimize their motions for each pathline of the optical flow. Wang et al. [126] first align frames to the same coordinate system by estimating camera motion, resize every frame spatially and temporally coherently, and then reconstruct resulting frames into the original coordinate system.

4.1.3. Warping

Building a mapping between each pair frame of the source video and the traget video is warping. Krahenbuhl et al. [10] proposes a realtime, pixel-accurate warping retargeting method with 2D variant of EWA splatting[127]. Zhang et al. [128] retarget videos in compressed-domain to save runtime. The video is first partially decoded, cropped with the saliency map, warped based on column-mesh to desired size, and finally re-encoded. Yan et al. [129] focus on eliminating jittery artifacts by considering spatial and temporal conherence simultaneously. [130] calculates saliency map based on gradient magnitude,

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

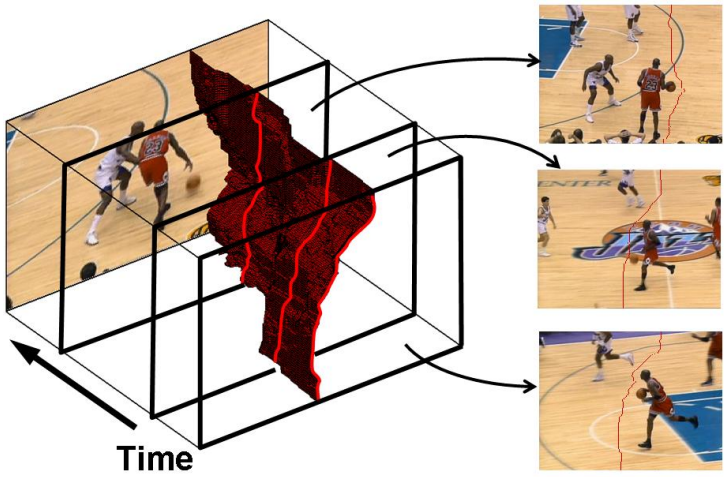

Figure 5. 3D cube and 2D seam manifolds[136].

face detection and motion detection, and treats the mapping betwwen source and retargeted pixels as a sparse linear system solved by a least squares algorithm. [131] points that saliency map with motion information is the cause of waving and squeezing artifacts. To avoid such artifacts, it takes motion into consideration, crops temporally-recurring contents, and warps homogeneous regions to mask deformations and preserve motion. Nie et al. [132] propose an interactive retargeting system that warps video using mean value coordinate warping method, and refines results by summarizing the temporal output based on patch to eliminate distoration. Consequently, this algorithm supports video summarization, completion, and reshuffling. To avoid repeated computation, Zhang et al. [133] compute the importance map for each pixel of every frame, and calculate cumulative shrinkability maps for the x and y directions and store them. Given any target resolution, it can quickly warps videos according to the shrinkability maps. Liu et al. [134] warp stereo 3D videos with disparity constraints. And Shao et al. [135] attempt to warp multi-view videos with depth.

4.1.4. Seam Carving

Seam carving reduces or expands image size by carving out or inserting seams[137]. For 1D seam, the image energy function in [137]:

$$e_{HoG} = \frac{|\frac{\partial}{\partial x}I| + |\frac{\partial}{\partial y}I|}{max(HoG(I(x,y)))} \tag{5}$$

Where I is an $n \times m$image, HoG is from [52]. And a vertical seam is:

$$\mathbf{s^x} = \{s_i^x\}_{i=1}^n = (x(i), i)_{i=1}^n, s.t. \forall i, |x(i) - x(i-1)| \leq k \tag{6}$$

where $k \in R$, and x is a mapping $x : [1, ..., n] \rightarrow [1, ..., m]$. To reduce image size, remove the seam with the least energy loss; To enlarge size, find the seam with the largest energy and average pixels of the seam with their neighbors.

Rubinstein et al. [136] extend 1D seams from 2D images to 2D seam manifolds from 3D space-time volumes of videos as shown in Figure 5. Unlike 1D seams that are calculated by dynamic programming, it solves the 2D seams by finding a minimal cut in the graph. Additionally, it proposes a novel energy function that emphasizes on the energy that caused by removing seams. [138] optimizes backward and forward energy jointly, and proposes isosurface protection and to encode the opacity transfer function. Kaur et al. [139] use Kalman filter to optimize the seam carving, which is theoretical simple and hardware friendly. Hsin et al. [140] manipulate saliency histogram, and propose saliency histogram equalisation-seam carving (SHE-SC) algorithm. They retarget the first frame of a video using SHE-SC, and adapt this algorithm for the rest frame according to their difference

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

from the previous frame. As 3D seam carving is of high computation cost, Furuta et al. [141] propose to find a suboptimal volume seam carving using multi-pass dynamic programming.

Retargeting stereo videos should follow three guidelines: keeping temporal coherence, preventing depth distortion, and minimizing shape distortions of the retargeted video[142]. Nguyen et al. [142] propose to segment stereo videos into groups of frames according to energy cost that includes saliency and stereoscopic confidence, and the seam carving within a group is fixed. It calcutes the left seam carving first. Refer to [120] for more works on video retargeting.

4.2. Video Summary

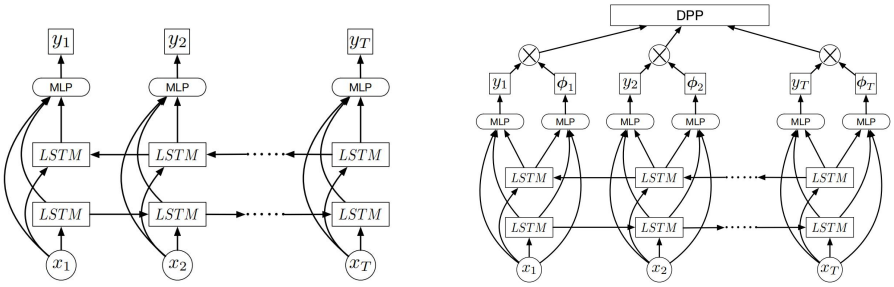

Figure 6. vsLSTM and dppLSTM from [143].

There are several solutions to reduce the size of videos. One is reducing frame rate by drop frames, which is called hyperlapse. One is extracting one or multiple keyframes and generating still images. And one is selecting saliency subshots and outputing a shortened video clips. In all, it should follow two main criteria: 1) preserving as much information as possible; 2) reducing the number of frames or sizes as much as possible. Thus, video summary is to seek a balance between information volume and output sizes according to different scene needs. Similar to the editing workflow above, video summary usually consists of segmentation, scoring the importance of each frame or shot, selecting the top significant frames or shots and alternatively output optimization and rendering. For certain videos, pre-processing is a must. For example, generating hyperlapse videos real-time for hand-held cameras requires stablization[87].

Segmentation is a key step. Zhao et al. [144] evenly segment input videos into clips of 50 frames to speed up. Poleg et al. [145] propose to use a novel Cumulative Displacement Curves to segment egocentric videos involing complex motion, and proves that integrated motion vectors work better than instantaneous motion vectors. That Calculating color histogram for frames, and detecting shot boundary when the differences between two adjacent frames is below some threshold, is a simple solution[146]. Potapov et al. [147] propose Kernel Temporal Segmentation (KTS) algorithm. Pavel et al. [13] segment video according to shot changes and the saliency maps. Abdelati Malek Amel et al. [148] detect shot boundaries by calculating the motion intensity using adaptive rood pattern search algorithm between two frames through the whole video and then deciding the threshold. Luo et al. [149] segment videos based on camera motion, eg. pan, zoom, pause. Zhang et al. [143] proposes two LSTM networks to model temporal dependency among frames and extracts keyframes or key subshots. The structures of vsLSTM and dppLSTM is shown in Figure 6.

Evaluating the importance of frames and shots varies a lot. Zhou et al. [150] propose time-mapping, namely transfroming high-frame-rate video to low-frame-rate video. This paper comes up a novel saliency method, a re-timing technique to temporally resample based on frame importance generated by that saliency method, and presents two new temporal filters (adaptive box filter and saliency-based motion-blur filter) to enhance the rendering of salient motion also generated by that saliency method. The novel saliency method is bottom-to-up, and utilizes sentimatic segmengtation and optical flow. The results

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

show that saliency-based motion-blur filter works best. Sun et al. [95] propose a pair-wise ranking model that learns from online videos and scores the highlightness of video clips without constraints. LiveLight[144] first scans and segments the input video based on dictionary. It builds a dictionary of video segments by add the new segment that could not be sparsely reconstructed[151] using present dictionary. Li et al. [152] also propose a dictionary learning approach to segment videos that considering reconstruction loss, group sparsity regularization, patch-level and frame-level structure preservation regularizer.

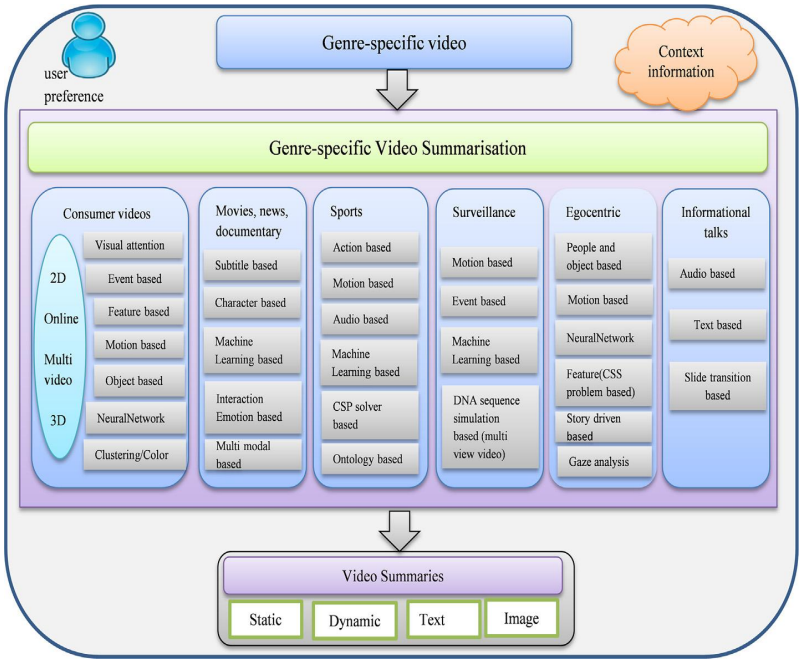

Figure 7. Work flow of [153].

The outputs can be a keyframe, multiple keyframes(static story boards)[154][93], a static storyboarding[155] or a shorter video clip[95]. Many classical video summary works has been concluded by Truong et al. [156]. Since 2006, more deep learning based methods are proposed to summarize videos. Pavel[19] summarizes videos based on their lines. Cong et al. [50] treat the keyframe extraction and video skim problem as a dictionary selection problem. And Hamza proposes to extract keyframes from wireless capsule endoscopy[93]. [92] proposes a novel human detection and tracking method based on poselet and a new saliency detector trained with gaze data. Gong et al. [157] propose sequential determinantal point process (seqDPP) to summarize videos in a supervised manner. It also provides evaluation metrics. Given two summaries, figure out the matched frames whose visual distance is under a threshold, and compute their precision, recall and F-score. And it synthesizes a ground truth summary per video by maximizing the F-score between ground truth and multiple human-annotated ones greedily.

For different genre videos, the methods to summarize them are also various. See Figure 7. Let us take egocentirc videos as an example. Wearable video cameras are first introduced by Steve Mann in 1998[158]. In 2001, Aizawa et al. proposed a system to summarize wearable videos[30] with brain wave cues. Since then, a lots of researches on egocentric video summary arise. Haung et al. [159] use support vector machine to summarize wearable videos. Ghosh focuses on discovering high-level saliency in egocentric video and forming a story board[160]. Lu et al. [8] come up with a new video editing solution that is story-driven and emphasizes causality between subshots. [161] uses web images as a prior to extract keyframes from user-generated videos of poor quality. [162] selects superframes according to their interestingness[163] and proposes SumMe benchmark. And Gygli further improves summary algorithm in a supervised way and jointly optimizes for multiple objectives[164]. In another work[165], Gygli emphasizes on

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

stablizing egocentric videos and utilizes their shifts to turn 2D videos into stereo. [166] proposes a series of graph based algorithm to detect the shoy boundary, select keyframes, capture the characteristics of a frame, extract features, and cluster keyframes. Xiong et al. [167] use web images as prior to detect the snap points in egocentric videos. Refer to [168] for more reviews on egocentric videos. Apart from egocentric videos, Zhou et al. [169] ultilize human face detection and tracking method to address the character-oriented summarization. Whereas Mindek et al. [170] summrize multiplayer games of 3D scenes based on game rules. More reviews on different genre video summarization can be found in [153].

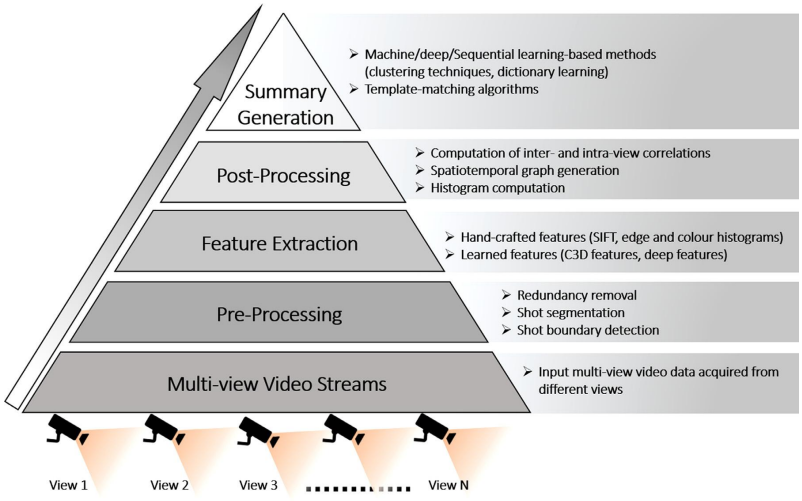

Figure 8. Overview of multi-view video summary[171].

Except single input video, there are also practical need of video summary about multiple videos. Figure 8 shows the general workflow of multi-view video summarization system. Fu et al. [6] firstly study the multi-view video summarization problem systematically and treats the multi-view video summary problem as a graph labeling task. It segments videos into shots, and computes their important scores based on low-level features(color histogram feature, edge histogram feature and wavelet feature) and high-level features(faces detected by Viola-Jones face detector[54]). By slection, a spatio-temporal shot graph of import shots is constructed, whose edges represent the similarity between nodes. Shots are clustered by randon walks, and multi-objective optimization. More works on multi-view video summary could be found in [171]. Meng et al. [172] propose centroid co-regularization (MSDS-CC) method to select representive visual elements. Whereas Li et al. [154] use SVM algorithm to abstract keyframes and further reduces keyframes with rough set. De et al. [173] aim at reducing inclusion of patches in the result summary videos for fixed-viewpoint multiple cameras by optimal reconstruction. Kuanar et al. [174] also regard multi-view video summarization problem as a graph-theoretic one. Apart from regular steps such as segmentation and feature extraction, it uses Gaussian entropy to drop redundant frames, ultilizes bipartite graph matching to calculate inter-view dependencies, and applies optimum-path forest algorithm to cluster keyframes. Wang et al. [175] learn metrics and output keyframes rather than videos. To reduce the compression and transmission power consumption of wireless video sensors, Ou et al. [5] design a simple but useful algorithm to summarize multi-view videos. It needs to be on-line, with low computational complexity and memory requirements. Additionally, as it is for multi-view sensors networks, the communication overhead between nodes is required to be as little as possible. This algorithm consists of two stages: intra-view stage and inter-view stage. In the intra-view stage, frame features are extracted by MPEG-7 color layout descriptor[46], clustered using simplified Gaussian mixture model(GMM), and the frames with smaller GMM weights and larger variances are selected. In the inter-view stage, only one frame of the same event is kept. [176] also considers the intra- and inter-correlations and focus

 doi:10.20944/preprints202201.0016.v1

on realizing sparsity. Zhong et al. [177] use a hypergraph based dominant set clustering method to locate keyframes, and utilizes web images to further reduce redundancy.

Apart from multi-view videos of overlapping view-of-field, Chu et al. [4] abstract visual co-occurrence shots from lots of videos of a certain topic without constraints. It segments videos by the color changes between two frames, and extracts frame features using CENTRIST[47], VLFeat[48] and HSV color moments[50]. The shots are constructed as a bipartite graph. And it proposes Maximal Biclique Finding algorithm to discard the shots that appear only within a single video.

4.3. Video Synposis

As video summarization is to extract consecutive or inconsecutive frames from an original video, video synposis further modifies the extracted frames. Besides, shifting interested activities in the time domain is a choice to further condense the inputs. Irani et al. firstly come up with generate synposis for video retrival[178]. Alex et al. [179] are the first one to come up with this concept. Video synopsis reaaranges moving objects at different time and locations at the same frames while keeping their locations unchanged. The workflow it proposes is as below: activity generation, tube rearrangement, background generation, object stitching, solving energy cost function within a given time interval by simulated annealing[180]. Obviously, the keypoint of video synopsis lies in finding optimal temporal positions of selected activities, compared to keyframe extraction. This work has many followers. For example, Pritch et al. [181] do a similar work for web cameras, and it could generate synopsis of limited length for a given time interval. And Kemal et al. give a thorough review on video synopsis about methodoloy[182]. Sun et al. [91] extract the saliency person performing an action and generates a photo montage.

Multi-view video synposis is also a explored problem. Mahapatra et al. [183] present a solution to generate multi-view video synposis consising of 5 steps: common background creation, common plane correspondence, object detection, action recognition and dynamic video synposis. Common background is the top view representation of the multiple cameras, and can be obtained with the help of Google Map for outdoor or by modifiying previous draft sketch. Common plane correspondence is achieved by mapping all cameras to a common coordinate[184]. Object detection is implemented with [185]. Action recognition is realized by SimpleMKL[186]. An energy function in terms of information loss, collision and length of resulting synposis is optimized by Simulated Annealing[180]. Generating a storyboarding needs tracking, semantic segmengtation, extracting keyframes, extending frame layout, annotation layout, compositing and rendering[155].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

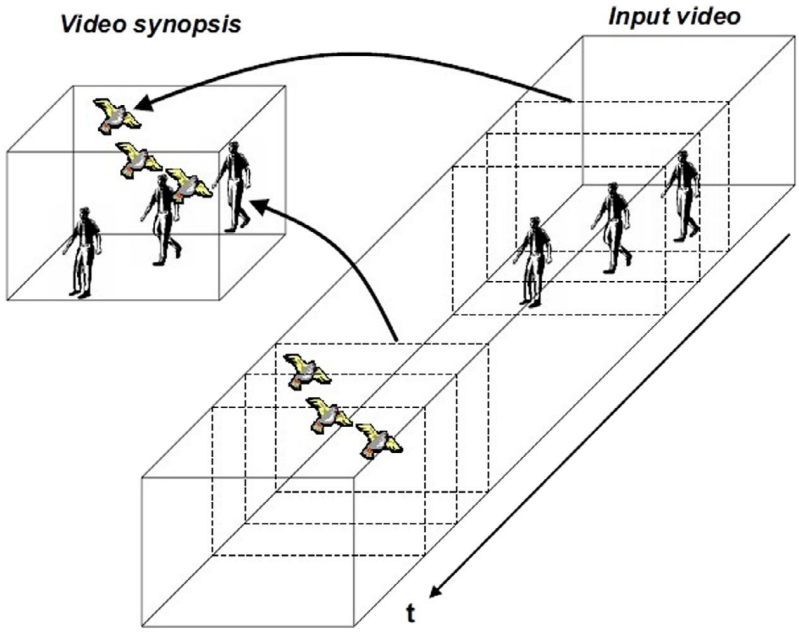

Figure 9. Video synposis[179].

Except for compression, video synposis can also function as a indexing method. Tang et al. [110] generate montages in which icons or sprites represent saliency events and function as indexes into the input video. Those sprites are extracted by building a Gaussian moxture model with conjugate priors for the background incrementally, and foreground elements are segmented with fast morphological operators.

4.4. Video Mosaics

Video masaics also rearrange the space-time volumes of input videos, but it manipulates a sequence of frames and does not modify the content of any frames. Rav et al. [187] propose Dynamosaics that sticks space-time volumes of 2D videos that is moving while recording to generate 360^o videos. And [188] seeks to stitch panoramas or activity synposis from web videos. It first filters videos by camera motions, moving objects and visual quality. Synthesize scene panoramas with fusion using feathering algorithm[189], or further add moving objects on the panoramas.

4.5. Special Effects

This section will introduce some interesting applications, that don't belong to previous subsections. Special effects range from obejct manipulation, style transferring, colorization, and so on.

Cliplet is a form between video and image for a small part of it is dynamic and the left is static. [16] first proposes the concept of cliplet. It comes up with an interactive system that support still, play, mirror and loop four idioms of iconic time-mapping fron input video to the target cliplet. After mapping, it refines results by automatic alignment, looping optimization and feathering, simultaneous matting and compositing, and Laplacian blending. Cinemagraphs is a special case of it for its movement is periodic. Bazin et al. [190] extract the target rigid object in input video or image, and users change its physical parameters. Then the new shape will be simulated and fit into original video.

Davis et al. [17] present a novel algorithm to find visual rhythm and beat in a video, and then warp the video with an audio track to form a dance video. Audio is processed by STFT to get its power spectrogram, onset envelope, tempogram and beats. Similarly, video's directogram is obtained by optical flow, and then generate impact envelope, visual tempogram and visual beats. Interpolation strategy is used to keep synchronization between audio and video. Except generating dance videos, it can also change the dance beats to fit another song. In this meaning, music-driven video editing is possible.

 doi:10.20944/preprints202201.0016.v1

Bai et al. [191] propose a semi-automated technique for selectively de-animating video to remove the large-scale motions of one or more objects. The user needs to draws three kinds of strokes: green strokes indicate which regions of the video should be de-animated; red strokes which regions should held static; blue strokes which should remain dynamic. This technique consists of two stages: warping and composition. Kanade-Lucas-Tomasi (KLT) tracking[192] is used to follow distinctive points in the input video and those tracks whose durations are less than 15% of the input are removed. Each input frame is divided into 64x32 rectilinear grid mesh. In initial warp, only anchor tracks are warped to minimize an energy function. And in refined warp, the output of the initial warp and floating tracks are used as input to solve the final energy function using least squares. At composing stage, graph cuts are used to perform Markov Random Field optimization.

Zhang et al. [193] propose vid2play, a system that learns the behaviors of tennis palyers with a huge volume annotated database. It could generates interactively controllable video sprites that behave and appear like professional players. Chang et al. [194] present a system that support object-level video editing. It segement objects and generates alpha mattes, and estimates 3D scene information based on structure from-motion (SfM) algorithm[195]. Users can apply 3D transformation to objects, duplicate objects, or even transfer objects across videos. Then the system will model 3D scene, render frames using sparse structure points and composite layers. Kasten et al. [196] propose to edit atlases of videos. Given a natural video with a coarse mask of interest obejcts, it estimates a set of atlases for backgorund and obejcts of interest. Users manipulate one or more atlases, like changing color, and adding texture. Then this algorithm estimates a mapping from each pixel in the video to a 2D point in each atlas, and its opacity, and propages the change consistently across the whole video. Coordinate-based Multilayer Perceptron (MLP) representation is used for mapping, atlases and alphas. This lagorithm is self-supervised and its loss function consists of rigidity loss, consistency loss and sparsity loss.

Fried et al. [197] present two method for puppet dubbing, one semi-automatic appearance-based and one fully automatic audio-based. Besides, the paper also proposes three guidelines for performing puppet speech: 1) each syllable in speech should match to one closed-open-closed segment of puppet lip motions, which is called visual syllables; 2) Lips should be still and closed when the puppet is not speaking, called visual silence syllables; 3) in rapid speech sequences, several spoken syllables may correspond to a single visual syllable. Inputs of the two methods are a given puppet video and a piece of new speech audio whose length is shorter than that of the video. Two approaches both can be divided in 4 steps. Segment the new speech audio track into a sequence of syllables by transcripting it into text with closed captions from YouTube, aligning the transcript to the audio using P2FA, combining the phonemes into syllables and merging short syllables. Segment the puppet video into a sequence of visual syllables using appearance-based or audio-based methods. In the appearance-based method, frames are classified into three categories: open-mouth, closed-mouth and invalid by a network based on pre-trained GoogLeNet or by hand or both. While in the audio-based method, visual syllables correspond to that of the first step. As for alignment audio syllables to visual syllables, three basic guidelines are 1) silence matches to silence; 2) non-silence matches to non-silence; 3) syllable lengths are similar as possible. Then solve a variant of dynamic time warping. At the last step, retime audio syllables using Waveform Similarity Overlap-add, retime visual syllables using nearest-neighbor sampling or optocal flow interpolation and retime audio and visual together by setting new length as their geometric mean. This paper declares their results in supplemental materials, but I could not find them. After all, puppet dubbing is much easier than human dubbing.

Video restoration also divided into special effects here. Lu et al. [198] use both first order and second order nonlocal regularization terms to restore videos of poor quality. Li et al. [199] propose a multiplanar autoregressive model to exploit the correlation in cross-dimensional planes of the group of similar patches of neighboring frames, and a joint multiplanar AR and low-rank based algorithm reconstructs the group. Finally, Markov

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

Random Field smoothes the temporally adjacent patches. Bai et al. [200] accelerate Monte Carlo simulation denoiser with the help of GPU.

Table 1. Datasets

Datasets in video editing domain	Description	Reference
Swedish leaf	images of leaves	[201]
67-class indoor scene	indoor images	[202]
Natural-Color	colorful images	[203]
SUN	scene photos	[204]
Food101	food images	[205]
26-scene	360^o panoramic scene images	[206]
ImageCLEF[10]	image retrival	
CMU-Mocap[11]	Motion Capture Database	
8 sports event	sports videos from web	[207]
Sports-360	360^o sports videos	[15]
Pano2Vid	360^o web videos	[14]
salientMontages[12]	unconstrained videos	[91]
Plenary session	mute and sound video saliency	[208]
MUFVET[13]	multi-face video seliency	[209]
VR gaze	VR saliency	[26]
LEDOV	video saliency	[57]
SumMe	video summary	[162]
TVSum50	video summary	[75]
Family outing	egocentric videos	[92]
PETS2009	multi-view videos	[210]
MED-summaries	video summary	[147]
Personal video	consumer videos and their keyframes	[149]
KTH IDOL		[211]
TRECVID[14]	video retrival	
Kodak[15]	consumer videos	[212]
TVFD	talking video frame drop	[105]
MoodSwings Turk	musical sentiment	[29]
EPFL stereo face	stereo face images	[213]
Five-task	instruction video	[77]

5. Datasets and Metrics

This section mainly give a brief review on datasets and evaluation metrics used in editing algorithms.

5.1. Datasets

Standard datasets greatly free researchers from labouring data collection and clean jobs. It saves time and money costs, and provoides a benchmark for the algorithms of this domain. Here, we sort 29 datasets from two main sources: challenges and researchers. Table 1 lists out the collected datasets. Sports-360[15] consisting of 342 360^o videos. They belongs to five sports categories: basketball, parkour, BMX, skateboarding, and dance. The videos of Pano2Vid[14] are collected from Youtube. Jiang et al. [57] builds LEDOV consisting of 32 subjects' fixations on 538 videos at least 720P resolution in 158 sub-categories. Hodosh et al. [214] establish a corpus of images with 5 simple captions. Swedish leaf[201], KTH IDOL[211], 15-class scene category[215], 8-class sports event[207], and 67-class indoor scene recognition[202] are image datasets for feature extraction. Alexander et al. [212] build

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

a Kodak' consumer video dataset of 25 concepts. The videos are from users and Youtube, and the concept labels are annotated by hand. The concepts includes activities(dancing, singing), occasions(wedding, birthday and so on), scene, object, people and sound. Luo et al. [149] select 100 video clips from Kodak' consumer video dataset and annotates their keyframes. Xiao et al. [206] construct a dataset of 360^o panoramic images in 26 place categories to train Support Vector Machine to learn place category and scene viewpoint of images, as well as symmtry of objects.

5.2. Evaluation Metrics

Evaluation metrics play an important role in algorithm designing for it gives feedbacks and suggests the drawbacks of the testing solution. There are several metrics commonly used in video editing domain, such as the degree of its automatic, speed, memory requirements. Precision, recall and F-measure(eg. F1-score), accuracy, CC score are popular metrics [7]. The definition of F1-score is:

$$\text{F1-score} = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{7}$$

It still is difficult to evaluate quality of generated videos, as discussed above that beauty and logic are hard to programme. Even though, researchers provide two solutions. One is user study. User feedback is the main metric. Su et al. [12] collect human visual saliency while they are watching output videos as the index to reflect how well the proposed algorithm works. Viewers' perferences are also a kind of feedback[13][19][85][8][7][102][98][44].

The other solution is to define metrics according to specific experiments. Zhang defines precision(P) and recall(R) as equation(2) for video summary[143]:

$$\text{P} = \frac{\text{overlapped duration of A and B}}{\text{duration of A}}, \text{R} = \frac{\text{overlapped duration of A and B}}{\text{duration of B}} \tag{8}$$

where A, B are video clips. Su et al. [14] define two classes of metrics. One is HumanCam-based and includes distinguishability, HumanCam-likeness and transferability. The other is HumanEdit-based metrics consisting of mean cosine similarity, frame pooling and mean overlap. Davis et al. [17] come up with *synchro-saliency* that measures the synchronization of visual and audible events. Potapov gives a method to evaluate the importance of video segment by considering whether the segment contains evidence of the given event category[147]. Sener et al. [33] use intersection over union(IOU) and mean average precision(mAP) to evaluate temporal segmentation:

$$IOU = \frac{1}{N} \sum_{i=1}^{N} \frac{\tau_i^* \cap \tau_i^{'}}{\tau_i^* \cup \tau_i^{'}} \tag{9}$$

where N is the number of segments, τ_i^* is the ground truth segment and $\tau_i^{'}$ is the predicted segment. To adopt to unsupervised algorithms, Sener utilizes cluster similarity measure that gives ground truth in brute-force searching manner. Liu et al. [216] propose metric to measure the image retargeting quality that is based on SIFT[49].

6. Image Editing

Though our focus is video editing, image editing is still a valuable research field worth exploring for the algorithms designed for images might can be transplanned to video domain. Here we will introduce recent researches on image editing, but we do not introduce the algorithms that generate images from scratch, like rendering[217].

To enhance the quality of videos is a valuable problem for a lot of images and videos have a long history. Super-resolution is not simply scaling images by interpolation, but emphasizing on improving its quality. It is an underdetermined inverse problem. Dong et al. [218] use deep CNN to learn the mapping from the low-resolution image to high-resolution one. [219][220] give a more detailed review on the super-resolution techniques.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 January 2022 doi:10.20944/preprints202201.0016.v1

Figure 10. [Left] original black white image; [Right] colorized result from [226].

Yu et al. [221] propose a noise prior learner NEGAN that denoises, inpaints and colorize legacy photos. More images denoise works utilizing deep network can be found in [222].

Arar et al. [223] apply seam carving in the feature map of the input image, and then reconstructs resulting image with CNN. Retargeting a pair of stereo images need to keep their piexl pairs and stereo structure unchanges by jointly optimization[224]. Dawei et al. [225] use seam carving and warping to retarget a pair of stereoscopic images utilizing their disparity consistency. Colorization (as shown in Figure 10) and style transferring is a prevalent field. Cheng et al. [227] first explore colorization with CNNs. And $(DE)^2CO$ uses deep learning network firstly[228]. Zhang et al. [229] propose a automatic colorization solution that treats it as a classification task and tries to increase the diversity of resulting colors. Some work need inputs from users. For example, [230] colorizes the grayscale images with a CNN trained on large datasets and sparse hints from users. Different from previous colorization methods that focus on the entire image, Su et al. [226] propose to fuse the object-level and image-level features obtained from two models to determine the final color of each object. Objects are detected by Mask R-CNN[231]and are cropped. The instance model, image-level models[230] and a fusion module are trained in three steps. More reviews on AI colorization can be found in [203][232]. Style transfer is extracting a texture from the source image domain and transfer it to the target image domain using a deep neural network. Jiang et al. [233] present a novel Ghost module into the GANILLA architecture[234] to learn and transfer the styles of images.

Figure 11. Soft Scissors from [235].

Wang et al. [235] propose an interactive tool for extracting alpha mattes of foreground objects, namely, Soft Scissors that combines incremental matte estimation, incremental foreground color estimation, intelligent user interface and robust matting algorithm so that could run in realtime and efficiently. While Kim et al. [236] grab several photo streams of the same theme, aligns them with similarity, and cosegments the shared regions of the aligned images based on image graph jointly. This work aims at finding the common patterns among a sea of web images.

To avoid repetitive efforts, Grabler et al. [237] propose a novel system to generate photo manipulation tutorials and content-dependent macros by recording a demonstration. It consists of demonstration recorder, image labeler and tutorial generator. Demostration recorder record all changes in the interface and the resulting changes in the application state and they will be grouped according to parameters and operations later. Image labeler leverages existing computer vision-based recognition technique to label semantically import regions in images. It generates text description in a fixed style with grouped changes, sceenshot annotations and text generation according to some guidelines, like step-by-step, succinct, text and images combination and grid-based layout.

Some works make a step towards image understanding. For example, given an image with several simple captions, Hodosh tries to resolve the entity coreference[214]. However, it only achieves a precision of 46.0% with F-score 49.0%. [163] investigates the interestingness of images.

7. Conclusion

7.1. Summary

Video editing starts from recent 20 years, and grows rapidly with the help of computer vision, natural language processing and other disciplines. We have given a clear introduction about the development history of AI video editing above: dividing editing tasks into different groups according to the number of input tracks, target output, features used, and the kinds of input videos. Common tools and datasets are sorted. As we can see, video editing systems evolves from simple to complex, from single input to multiple, from single feature to multiple. The range of input video domains is wide, from sports game to surveillance, from film to social platform short videos.

However, video editing still does not realize fully automatically. Now editing algorithms heavily rely on perfect inputs, and even pre-process data by human. The features extracted from audio, image and the temporal dependency of consecutive frames, and subtitle, are not easy understanding for computer machine, or enough for algorithm. Besides, designing editing plans that are logically consistent is very hard. Producing an amazing edited video is further impossible today. We hope this survey could provide convenience to researchers and promote video editing development.

7.2. Future Work

Even though video editing has already obtained impressive improvements over decades, it is still not intelligent enough. The degree of automation is waiting increase. Hopefully, several reaseaches have appered and perform successfully, which could bring insights in AI video editing. More researches now focus on video understanding, which would greatly help intellegnet video editing. Ramanathan et al. [238] attempt to resolve names for each figures. Haurilet et al. [239] propose a method to label all character appearances in TV series only using subtiles. [240] also identifies characters in TV. Du et. al. [53] achieve great results on action, scene and object recognition. Jean-Baptiste Alayrac[241] tries to discover actions and corresponding object states in videos. Huang et al. [242] try to solve references in instructional videos. Then [243] proposes visual understanding task for the video domain, present a novel visual grounding model that is both reference-aware and weakly-supervised, and provides reference-grounding test set annotations for YouCook2II and RoboWatch instructional video benchmarks. Xia et al. [244] propose an online multi-modal searching machine(OMS) to search persons in videos with their features of face, body and voice. Similar work on person searching in video includes [245][246].

Video saliency give hints of what audiences want to see. Scherer et al. [208] investigate the impact of audiovisual features in the communication and judgement of politicians in zero acquaintance situations. Authors make several expriments to explore the disparity in the speakers' perception with and without audio using eye-tracking data and the influence of audio features, together with common visual features on the perception of the speakers' qualities. Jiang et al. [57] have three findings: 1) high correlation exits between objectness and human attention; 2) objects, especially, moving objects or parts are more attractive for human attention; 3) sliency maps are smoothly transited across frames. Based on those findings, an object-to-motion model OM-CNN is developed to learn motion features to predict intra-frame saliency and saliency-structured convolutional long short-term memory network SS-ConvLSTM is proposed to learn eye pixelwise transition across frames and center-bias for inter-frame saliency maps. Liu follows this work in [209], and also proposes a deep learning model to predict salient face with transition across frames in multiple-face videos and builds a new multiface video database. By analyzing the database, two findings

can be concluded: 1) faces accounting for 5% pixels draw about 80% attention in multiface videos and one face in each frame draws most subjects' attention; 2) humans tend to focus on the face close to the center of videos. While Xia et al. [59] explore saliency with user bias. [247] also investigates the video saliency.

8. Patents

Author Contributions: Conceptualization, Xinrong Zhang and Jiangtao Wen; formal analysis, Xinrong Zhang and Yanghao Li; investigation, Xinrong Zhang; resources, JiangtaoWen; writing—original draft preparation, Xinrong Zhang; writing—review and editing, Xinrong Zhang, Yanghao Li, Yuxing Han, and Jiangtao Wen; supervision, Jiangtao Wen; project administration, Yuxing Han; funding acquisition, Yuxing Han. All authors have read and agreed to the published version of the manuscript.

Funding: We acknowledge funding from the Shenzhen Science and Technology Program (Grant No.KQTD20190929172829742).

Conflicts of Interest: The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. The Sustainable Future of Video Entertainment **2020**.
2. Mukhopadhyay, S.; Smith, B. Passive Capture and Structuring of Lectures. Proceedings of the Seventh ACM International Conference on Multimedia (Part 1); Association for Computing Machinery: New York, NY, USA, 1999; MULTIMEDIA '99, p. 477–487. doi:10.1145/319463.319690.
3. Zhang, C.; Rui, Y.; Crawford, J.; He, L.W. An Automated End-to-End Lecture Capture and Broadcasting System. *ACM Trans. Multimedia Comput. Commun. Appl.* **2008**, *4*. doi:10.1145/1324287.1324293.
4. Chu, W.S.; Song, Y.; Jaimes, A. Video co-summarization: Video summarization by visual co-occurrence. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3584–3592. doi:10.1109/CVPR.2015.7298981.
5. Ou, S.H.; Lee, C.H.; Somayazulu, V.S.; Chen, Y.K.; Chien, S.Y. On-Line Multi-View Video Summarization for Wireless Video Sensor Network. *IEEE Journal of Selected Topics in Signal Processing* **2015**, *9*, 165–179. doi:10.1109/JSTSP.2014.2331916.
6. Fu, Y.; Guo, Y.; Zhu, Y.; Liu, F.; Song, C.; Zhou, Z.H. Multi-View Video Summarization. *IEEE Transactions on Multimedia* **2010**, *12*, 717–729. doi:10.1109/TMM.2010.2052025.
7. Panda, R.; Roy-Chowdhury, A.K. Multi-View Surveillance Video Summarization via Joint Embedding and Sparse Optimization. *IEEE Transactions on Multimedia* **2017**, *19*, 2010–2021. doi:10.1109/TMM.2017.2708981.
8. Lu, Z.; Grauman, K. Story-Driven Summarization for Egocentric Video. Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition; IEEE Computer Society: USA, 2013; CVPR '13, p. 2714–2721. doi:10.1109/CVPR.2013.350.
9. Liu, F.; Gleicher, M. Video retargeting: automating pan and scan. Proceedings of the 14th ACM international conference on Multimedia, 2006, pp. 241–250.
10. Krähenbühl, P.; Lang, M.; Hornung, A.; Gross, M. A System for Retargeting of Streaming Video. *ACM Trans. Graph.* **2009**, *28*, 1–10. doi:10.1145/1618452.1618472.
11. Sivic.; Zisserman. Video Google: a text retrieval approach to object matching in videos. Proceedings Ninth IEEE International Conference on Computer Vision, 2003, pp. 1470–1477 vol.2. doi:10.1109/ICCV.2003.1238663.
12. Su, Y.C.; Grauman, K. Making 360° Video Watchable in 2D: Learning Videography for Click Free Viewing. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1368–1376. doi:10.1109/CVPR.2017.150.
13. Pavel, A.; Hartmann, B.; Agrawala, M. Shot Orientation Controls for Interactive Cinematography with 360 Video. Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology; Association for Computing Machinery: New York, NY, USA, 2017; UIST '17, p. 289–297. doi:10.1145/3126594.3126636.
14. Su, Y.C.; Jayaraman, D.; Grauman, K. Pano2Vid: Automatic Cinematography for Watching 360° Videos. *ArXiv* **2017**, *abs/1612.02335*.
15. Hu, H.N.; Lin, Y.C.; Liu, M.Y.; Cheng, H.T.; Chang, Y.J.; Sun, M. Deep 360 Pilot: Learning a Deep Agent for Piloting through 360° Sports Videos. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* **2017**, pp. 1396–1405.
16. Joshi, N.; Mehta, S.; Drucker, S.; Stollnitz, E.; Hoppe, H.; Uyttendaele, M.; Cohen, M., Cliplets: Juxtaposing Still and Dynamic Imagery. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*; Association for Computing Machinery: New York, NY, USA, 2012; p. 251–260.
17. Davis, A.; Agrawala, M. Visual Rhythm and Beat. *ACM Trans. Graph.* **2018**, *37*. doi:10.1145/3197517.3201371.
18. Xia, H.; Jacobs, J.; Agrawala, M. Crosscast: Adding Visuals to Audio Travel Podcasts. Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology; Association for Computing Machinery: New York, NY, USA, 2020; UIST '20, p. 735–746. doi:10.1145/3379337.3415882.

19. Pavel, A.; Goldman, D.B.; Hartmann, B.; Agrawala, M. SceneSkim: Searching and Browsing Movies Using Synchronized Captions, Scripts and Plot Summaries. Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology; Association for Computing Machinery: New York, NY, USA, 2015; UIST '15, p. 181–190. doi:10.1145/2807442.2807502.
20. Leake, M.; Davis, A.; Truong, A.; Agrawala, M. Computational Video Editing for Dialogue-Driven Scenes. *ACM Trans. Graph.* **2017**, *36*. doi:10.1145/3072959.3073653.
21. Faghri, F.; Fleet, D.J.; Kiros, J.R.; Fidler, S. VSE++: Improving Visual-Semantic Embeddings with Hard Negatives, 2018, [arXiv:cs.LG/1707.05612].
22. Mathe, S.; Sminchisescu, C. Actions in the Eye: Dynamic Gaze Datasets and Learnt Saliency Models for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2015**, *37*, 1408–1424. doi:10.1109/TPAMI.2014.2366154.
23. Xu, Y.; Gao, S.; Wu, J.; Li, N.; Yu, J. Personalized Saliency and Its Prediction. *IEEE Transactions on Pattern Analysis and Machine Int elligence* **2019**, *41*, 2975–2989. doi:10.1109/TPAMI.2018.2866563.
24. Wang, W.; Shen, J.; Xie, J.; Cheng, M.M.; Ling, H.; Borji, A. Revisiting Video Saliency Prediction in the Deep Learning Era. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2021**, *43*, 220–237. doi:10.1109/TPAMI.2019.2924417.
25. Serrano, A.; Sitzmann, V.; Ruiz-Borau, J.; Wetzstein, G.; Gutierrez, D.; Masia, B. Movie Editing and Cognitive Event Segmentation in Virtual Reality Video. *ACM Trans. Graph.* **2017**, *36*. doi:10.1145/3072959.3073668.
26. Sitzmann, V.; Serrano, A.; Pavel, A.; Agrawala, M.; Gutierrez, D.; Masia, B.; Wetzstein, G. Saliency in VR: How Do People Explore Virtual Environments? *IEEE Transactions on Visualization and Computer Graphics* **2018**, *24*, 1633–1642. doi:10.1109/TVCG.2018.2793599.
27. Rubin, S.; Berthouzoz, F.; Mysore, G.; Li, W.; Agrawala, M. UnderScore: Musical Underlays for Audio Stories. Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology; Association for Computing Machinery: New York, NY, USA, 2012; UIST '12, p. 359–366. doi:10.1145/2380116.2380163.
28. Rubin, S.; Berthouzoz, F.; Mysore, G.J.; Li, W.; Agrawala, M. Content-based tools for editing audio stories. *Proceedings of the 26th annual ACM symposium on User interface software and technology* **2013**.
29. Rubin, S.; Agrawala, M. Generating Emotionally Relevant Musical Scores for Audio Stories. Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology; Association for Computing Machinery: New York, NY, USA, 2014; UIST '14, p. 439–448. doi:10.1145/2642918.2647406.
30. Aizawa, K.; Ishijima, K.; Shiina, M. Summarizing wearable video. Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205), 2001, Vol. 3, pp. 398–401 vol.3. doi:10.1109/ICIP.2001.958135.
31. Rui, Y.; Gupta, A.; Acero, A. Automatically extracting highlights for TV Baseball programs. *Proceedings of the eighth ACM international conference on Multimedia* **2000**.
32. Wang, M.; Yang, G.W.; Hu, S.M.; Yau, S.T.; Shamir, A. Write-a-Video: Computational Video Montage from Themed Text. *ACM Trans. Graph.* **2019**, *38*. doi:10.1145/3355089.3356520.
33. Sener, O.; Zamir, A.R.; Savarese, S.; Saxena, A. Unsupervised Semantic Parsing of Video Collections. 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 4480–4488. doi:10.1109/ICCV.2015.509.
34. Truong, A.; Chi, P.; Salesin, D.; Essa, I.; Agrawala, M., Automatic Generation of Two-Level Hierarchical Tutorials from Instructional Makeup Videos. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*; Association for Computing Machinery: New York, NY, USA, 2021.
35. Xiong, Z.; Radhakrishnan, R.; Divakaran, A.; Huang, T. Effective and efficient sports highlights extraction using the minimum description length criterion in selecting GMM structures [audio classification]. 2004 IEEE International Conference on Multimedia and Expo (ICME) (IEEE Cat. No.04TH8763), 2004, Vol. 3, pp. 1947–1950 Vol.3. doi:10.1109/ICME.2004.1394642.
36. Cheng, C.C.; Hsu, C.T. Fusion of audio and motion information on HMM-based highlight extraction for baseball games. *IEEE Transactions on Multimedia* **2006**, *8*, 585–599. doi:10.1109/TMM.2006.870726.
37. Li, B.; Sezan, I. Semantic sports video analysis: approaches and new applications. Proceedings 2003 International Conference on Image Processing (Cat. No.03CH37429), 2003, Vol. 1, pp. I–17. doi:10.1109/ICIP.2003.1246887.
38. Heck, R.; Wallick, M.; Gleicher, M. Virtual Videography. *ACM Trans. Multimedia Comput. Commun. Appl.* **2007**, *3*, 4–es. doi:10.1145/1198302.1198306.
39. Truong, A.; Berthouzoz, F.; Li, W.; Agrawala, M. QuickCut: An Interactive Tool for Editing Narrated Video. Proceedings of the 29th Annual Symposium on User Interface Software and Technology; Association for Computing Machinery: New York, NY, USA, 2016; UIST '16, p. 497–507. doi:10.1145/2984511.2984569.
40. Leake, M.; Shin, H.; Kim, J.; Agrawala, M. Generating Audio-Visual Slideshows from Text Articles Using Word Concreteness. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* **2020**.
41. Xu, R.; Cao, J.; Wang, M.; Chen, J.; Zhou, H.; Zeng, Y.; Wang, Y.; Chen, L.; Yin, X.; Zhang, X.; Jiang, S.; Wang, Y.; Li, L. Xiaomingbot: A Multilingual Robot News Reporter. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations; Association for Computational Linguistics: Online, 2020; pp. 1–8. doi:10.18653/v1/2020.acl-demos.1.
42. Shin, H.V.; Li, W.; Durand, F. Dynamic Authoring of Audio with Linked Scripts. Proceedings of the 29th Annual Symposium on User Interface Software and Technology; Association for Computing Machinery: New York, NY, USA, 2016; UIST '16, p. 509–516. doi:10.1145/2984511.2984561.

43. Ellis, D.P.; Poliner, G.E. Identifying 'Cover Songs' with Chroma Features and Dynamic Programming Beat Tracking. 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, 2007, Vol. 4, pp. IV–1429–IV–1432. doi:10.1109/ICASSP.2007.367348.
44. Shin, H.V.; Berthouzoz, F.; Li, W.; Durand, F. Visual Transcripts: Lecture Notes from Blackboard-Style Lecture Videos. *ACM Trans. Graph.* **2015**, *34*. doi:10.1145/2816795.2818123.
45. Tamura, H.; Mori, S.; Yamawaki, T. Textural Features Corresponding to Visual Perception. *IEEE Transactions on Systems, Man, and Cybernetics* **1978**, *8*, 460–473. doi:10.1109/TSMC.1978.4309999.
46. Kasutani, E.; Yamada, A. The MPEG-7 color layout descriptor: a compact image feature description for high-speed image/video segment retrieval. Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205), 2001, Vol. 1, pp. 674–677 vol.1. doi:10.1109/ICIP.2001.959135.
47. Wu, J.; Rehg, J.M. CENTRIST: A Visual Descriptor for Scene Categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2011**, *33*, 1489–1501. doi:10.1109/TPAMI.2010.224.
48. Vedaldi, A.; Fulkerson, B. Vlfeat: An Open and Portable Library of Computer Vision Algorithms. Proceedings of the 18th ACM International Conference on Multimedia; Association for Computing Machinery: New York, NY, USA, 2010; MM '10, p. 1469–1472. doi:10.1145/1873951.1874249.
49. LoweDavid, G. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* **2004**.
50. Cong, Y.; Yuan, J.; Luo, J. Towards Scalable Summarization of Consumer Videos Via Sparse Dictionary Selection. *IEEE Transactions on Multimedia* **2012**, *14*, 66–75. doi:10.1109/TMM.2011.2166951.
51. Oliva, A.; Torralba, A. Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope. *International Journal of Computer Vision* **2004**, *42*, 145–175.
52. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005, Vol. 1, pp. 886–893 vol. 1. doi:10.1109/CVPR.2005.177.
53. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning spatiotemporal features with 3D convolutional networks. In ICCV, 2015.
54. Viola, P.; Jones, M. Rapid object detection using a boosted cascade of simple features. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 2001, Vol. 1, pp. I–I. doi:10.1109/CVPR.2001.990517.
55. Xu, M.; Ren, Y.; Wang, Z. Learning to Predict Saliency on Face Images. 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 3907–3915. doi:10.1109/ICCV.2015.445.
56. Liu, Y.; Zhang, S.; Xu, M.; He, X. Predicting Salient Face in Multiple-Face Videos. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 3224–3232. doi:10.1109/CVPR.2017.343.
57. Jiang, L.; Xu, M.; Liu, T.; Qiao, M.; Wang, Z. DeepVS: A Deep Learning Based Video Saliency Prediction Approach. ECCV, 2018.
58. Xia, J.; Tian, J.; Xing, J.; Cheng, J.; Zhang, J.; Wen, J.; Li, Z.; Lou, J.G. Social Data Assisted Multi-Modal Video Analysis For Saliency Detection. *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* **2020**, pp. 2278–2282.
59. Xia, J.; Tian, J.; Qiao, H.; Li, Y.; Wen, J.; Han, Y. Multimodal Video Saliency Analysis With User-Biased Information. 2020 IEEE International Conference on Multimedia and Expo (ICME), 2020, pp. 1–6. doi:10.1109/ICME46284.2020.9102908.
60. Park, H.S.; Jain, E.; Sheikh, Y. 3D Social Saliency from Head-Mounted Cameras. Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1; Curran Associates Inc.: Red Hook, NY, USA, 2012; NIPS'12, p. 422–430.
61. Xu, M.; Song, Y.; Wang, J.; Qiao, M.; Huo, L.; Wang, Z. Predicting Head Movement in Panoramic Video: A Deep Reinforcement Learning Approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2019**, *41*, 2693–2708. doi:10.1109/TPAMI.2018.2858783.
62. Parkhi, O.M.; Vedaldi, A.; Zisserman, A. Deep Face Recognition. BMVC, 2015.
63. Saragih, J.M.; Lucey, S.; Cohn, J.F. Face alignment through subspace constrained mean-shifts. 2009 IEEE 12th International Conference on Computer Vision, 2009, pp. 1034–1041. doi:10.1109/ICCV.2009.5459377.
64. Wei, S.E.; Ramakrishna, V.; Kanade, T.; Sheikh, Y. Convolutional Pose Machines, 2016, [arXiv:cs.CV/1602.00134].
65. Cao, Z.; Simon, T.; Wei, S.E.; Sheikh, Y. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields, 2017, [arXiv:cs.CV/1611.08050].
66. Bourdev, L.; Malik, J. Poselets: Body part detectors trained using 3D human pose annotations. 2009 IEEE 12th International Conference on Computer Vision, 2009, pp. 1365–1372. doi:10.1109/ICCV.2009.5459303.
67. Girdhar, R.; Gkioxari, G.; Torresani, L.; Paluri, M.; Tran, D. Detect-and-Track: Efficient Pose Estimation in Videos, 2018, [arXiv:cs.CV/1712.09184].
68. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, 2016, [arXiv:cs.CV/1506.01497].
69. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition, 2015, [arXiv:cs.CV/1409.1556].
70. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1–9. doi:10.1109/CVPR.2015.7298594.
71. Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; Darrell, T. Caffe: Convolutional Architecture for Fast Feature Embedding, 2014, [arXiv:cs.CV/1408.5093].

72. Jiang, Y.G.; Yang, J.; Ngo, C.W.; Hauptmann, A.G. Representations of Keypoint-Based Semantic Concept Detection: A Comprehensive Study. *IEEE Transactions on Multimedia* **2010**, *12*, 42–53. doi:10.1109/TMM.2009.2036235.
73. Zhu, Y.; Chen, Z.; Wu, F. Multimodal Deep Denoise Framework for Affective Video Content Analysis. Proceedings of the 27th ACM International Conference on Multimedia; Association for Computing Machinery: New York, NY, USA, 2019; MM '19, p. 130–138. doi:10.1145/3343031.3350997.
74. Pavel, A.; Reed, C.; Hartmann, B.; Agrawala, M. Video Digests: A Browsable, Skimmable Format for Informational Lecture Videos. Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology; Association for Computing Machinery: New York, NY, USA, 2014; UIST '14, p. 573–582. doi:10.1145/2642918.2647400.
75. Song, Y.; Vallmitjana, J.; Stent, A.; Jaimes, A. TVSum: Summarizing web videos using titles. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 5179–5187. doi:10.1109/CVPR.2015.7299154.
76. de Marneffe, M.C.; MacCartney, B.; Manning, C.D. Generating Typed Dependency Parses from Phrase Structure Parses. Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06); European Language Resources Association (ELRA): Genoa, Italy, 2006.
77. Alayrac, J.B.; Bojanowski, P.; Agrawal, N.; Sivic, J.; Laptev, I.; Lacoste-Julien, S. Unsupervised Learning from Narrated Instruction Videos. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* **2016**, pp. 4575–4583.
78. Xu, W.; Liu, X.; Gong, Y. Document Clustering Based on Non-Negative Matrix Factorization. Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval; Association for Computing Machinery: New York, NY, USA, 2003; SIGIR '03, p. 267–273. doi:10.1145/860435.860485.
79. Malmaud, J.; Huang, J.; Rathod, V.; Johnston, N.; Rabinovich, A.; Murphy, K.P. What's Cookin'? Interpreting Cooking Videos using Text, Speech and Vision. NAACL, 2015.
80. van den Oord, A.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; Kavukcuoglu, K. WaveNet: A Generative Model for Raw Audio, 2016, [arXiv:cs.SD/1609.03499].
81. Truong, A.; Agrawala, M. A Tool for Navigating and Editing 360 Video of Social Conversations into Shareable Highlights. Proceedings of the 45th Graphics Interface Conference on Proceedings of Graphics Interface 2019; Canadian Human-Computer Communications Society: Waterloo, CAN, 2019; GI'19. doi:10.20380/GI2019.14.
82. Owens, A.; Efros, A.A. Audio-Visual Scene Analysis with Self-Supervised Multisensory Features, 2018, [arXiv:cs.CV/1804.03641].
83. Cour, T.; Sapp, B.; Nagle, A.; Taskar, B. Talking pictures: Temporal grouping and dialog-supervised person recognition. 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010, pp. 1014–1021. doi:10.1109/CVPR.2010.5540106.
84. Sivic, J.; Everingham, M.; Zisserman, A. "Who are you?" - Learning person specific classifiers from video. 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1145–1152. doi:10.1109/CVPR.2009.5206513.
85. Chi, P.Y.; Liu, J.; Linder, J.; Dontcheva, M.; Li, W.; Hartmann, B. DemoCut: Generating Concise Instructional Videos for Physical Demonstrations. Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology; Association for Computing Machinery: New York, NY, USA, 2013; UIST '13, p. 141–150. doi:10.1145/2501988.2502052.
86. Malmaud, J.; Wagner, E.; Chang, N.; Murphy, K. Cooking with Semantics. Proceedings of the ACL 2014 Workshop on Semantic Parsing; Association for Computational Linguistics: Baltimore, MD, 2014; pp. 33–38. doi:10.3115/v1/W14-2407.
87. Joshi, N.; Kienzle, W.; Toelle, M.; Uyttendaele, M.; Cohen, M.F. Real-Time Hyperlapse Creation via Optimal Frame Selection. *ACM Trans. Graph.* **2015**, *34*. doi:10.1145/2766954.
88. Fischler, M.A.; Bolles, R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* **1981**, *24*, 381–395. doi:10.1145/358669.358692.
89. Calonder, M.; Lepetit, V.; Strecha, C.; Fua, P. BRIEF: Binary Robust Independent Elementary Features. Computer Vision – ECCV 2010; Daniilidis, K.; Maragos, P.; Paragios, N., Eds.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2010; pp. 778–792.
90. Kopf, J.; Cohen, M.; Szeliski, R. First-person Hyperlapse Videos. ACM Transactions on Graphics (Proc. SIGGRAPH 2014), ACM Transactions on Graphics (Proc. SIGGRAPH 2014) ed. ACM - Association for Computing Machinery, 2014, Vol. 33.
91. Sun, M.; Farhadi, A.; Taskar, B.; Seitz, S. Summarizing Unconstrained Videos Using Salient Montages. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2017**, *39*, 2256–2269. doi:10.1109/TPAMI.2016.2623699.
92. Sun, M.; Farhadi, A.; Taskar, B.; Seitz, S. Salient Montages from Unconstrained Videos. Computer Vision – ECCV 2014; Springer International Publishing: Cham, 2014; pp. 472–488.
93. Hamza, R.; Muhammad, K.; Lv, Z.; Titouna, F. Secure Video Summarization Framework for Personalized Wireless Capsule Endoscopy. *Pervasive Mob. Comput.* **2017**, *41*, 436–450. doi:10.1016/j.pmcj.2017.03.011.
94. Hanjalic, A. Adaptive extraction of highlights from a sport video based on excitement modeling. *IEEE Transactions on Multimedia* **2005**, *7*, 1114–1122. doi:10.1109/TMM.2005.858397.
95. Sun, M.; Farhadi, A.; Seitz, S. Ranking Domain-Specific Highlights by Analyzing Edited Videos. Computer Vision – ECCV 2014; Fleet, D.; Pajdla, T.; Schiele, B.; Tuytelaars, T., Eds.; Springer International Publishing: Cham, 2014; pp. 787–802.
96. Galvane, Q.; Ronfard, R.; Lino, C.; Christie, M. Continuity Editing for 3D Animation. Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence. AAAI Press, 2015, AAAI'15, p. 753–761.
97. Jain, E.; Sheikh, Y.; Shamir, A.; Hodgins, J. Gaze-Driven Video Re-Editing. *ACM Trans. Graph.* **2015**, *34*. doi:10.1145/2699644.
98. Rachavarapu, K.K.; Kumar, M.; Gandhi, V.; Subramanian, R. Watch to Edit: Video Retargeting using Gaze. *Computer Graphics Forum* **2018**, *37*, 205–215, [https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13354]. doi:https://doi.org/10.1111/cgf.13354.

99. Khoenkaw, P.; Piamsa-nga, P. Automatic pan-and-scan algorithm for heterogeneous displays. *Multimedia Tools and Applications* **2014**, *74*. doi:10.1007/s11042-014-2308-4.
100. Ooi, W.T.; Smith, B.C.; Mukhopadhyay, S.; Chan, H.H.; Weiss, S.; Chiu, M. Dali multimedia software library. Multimedia Computing and Networking 1999; Kandlur, D.D.; Jeffay, K.; Roscoe, T., Eds. International Society for Optics and Photonics, SPIE, 1998, Vol. 3654, pp. 264 – 275. doi:10.1117/12.333816.
101. Gleicher, M.L.; Heck, R.M.; Wallick, M.N. A Framework for Virtual Videography. Proceedings of the 2nd International Symposium on Smart Graphics; Association for Computing Machinery: New York, NY, USA, 2002; SMARTGRAPH '02, p. 9–16. doi:10.1145/569005.569007.
102. Ranjan, A.; Birnholtz, J.; Balakrishnan, R. Improving Meeting Capture by Applying Television Production Principles with Audio and Motion Detection. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems; Association for Computing Machinery: New York, NY, USA, 2008; CHI '08, p. 227–236. doi:10.1145/1357054.1357095.
103. He, L.w.; Cohen, M.F.; Salesin, D.H. The Virtual Cinematographer: A Paradigm for Automatic Real-Time Camera Control and Directing. Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques; Association for Computing Machinery: New York, NY, USA, 1996; SIGGRAPH '96, p. 217–224. doi:10.1145/237170.237259.
104. Berthouzoz, F.; Li, W.; Agrawala, M. Tools for Placing Cuts and Transitions in Interview Video. *ACM Trans. Graph.* **2012**, *31*. doi:10.1145/2185520.2185563.
105. Cheng, H.; Guo, Y.; Yin, J.; Chen, H.; Wang, J.; Nie, L. Audio-driven Talking Video Frame Restoration. *IEEE Transactions on Multimedia* **2021**, pp. 1–1. doi:10.1109/TMM.2021.3118287.
106. Lai, W.S.; Huang, Y.; Joshi, N.; Buehler, C.; Yang, M.H.; Kang, S.B. Semantic-driven Generation of Hyperlapse from 360° Video. *ArXiv* **2017**, *abs/1703.10798*.
107. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation, 2015, [arXiv:cs.CV/1411.4038].
108. Bozdogan, H. Akaike's Information Criterion and Recent Developments in Information Complexity. *Journal of Mathematical Psychology* **2000**, *44*, 62–91. doi:https://doi.org/10.1006/jmps.1999.1277.
109. Liu, S.; Yuan, L.; Tan, P.; Sun, J. Bundled Camera Paths for Video Stabilization. *ACM Trans. Graph.* **2013**, *32*. doi:10.1145/2461912.2461995.
110. Tang, C.; Wang, O.; Liu, F.; Tan, P. Joint Stabilization and Direction of 360°;Videos. *ACM Trans. Graph.* **2019**, *38*. doi:10.1145/3211889.
111. Elhamifar, E.; Vidal, R. Sparse Subspace Clustering: Algorithm, Theory, and Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2013**, *35*, 2765–2781. doi:10.1109/TPAMI.2013.57.
112. He, R.; Tan, T.; Wang, L.; Zheng, W.S. l2, 1 Regularized correntropy for robust feature selection. 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 2504–2511. doi:10.1109/CVPR.2012.6247966.
113. He, R.; Zheng, W.S.; Tan, T.; Sun, Z. Half-Quadratic-Based Iterative Minimization for Robust Sparse Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2014**, *36*, 261–275. doi:10.1109/TPAMI.2013.102.
114. Arev, I.; Park, H.S.; Sheikh, Y.; Hodgins, J.; Shamir, A. Automatic Editing of Footage from Multiple Social Cameras. *ACM Trans. Graph.* **2014**, *33*. doi:10.1145/2601097.2601198.
115. Dale, K.; Shechtman, E.; Avidan, S.; Pfister, H. Multi-video browsing and summarization. 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2012, pp. 1–8. doi:10.1109/CVPRW.2012.6239253.
116. Huber, B.; Shin, H.; Russell, B.C.; Wang, O.; Mysore, G.J. B-Script: Transcript-based B-roll Video Editing with Recommendations. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* **2019**.
117. Wang, Z.; Li, J.; Jiang, Y.G. Story-driven Video Editing. *IEEE Transactions on Multimedia* **2021**, *23*, 4027–4036. doi:10.1109/TMM.2020.3037461.
118. Arandjelović, R.; Gronát, P.; Torii, A.; Pajdla, T.; Sivic, J. NetVLAD: CNN Architecture for Weakly Supervised Place Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2018**, *40*, 1437–1451.
119. Vaquero, D.A.; Turk, M.A.; Pulli, K.; Tico, M.; Gelfand, N. A survey of image retargeting techniques. Optical Engineering + Applications, 2010.
120. Kiess, J.; Kopf, S.; Guthier, B.; Effelsberg, W. A Survey on Content-Aware Image and Video Retargeting. *ACM Trans. Multimedia Comput. Commun. Appl.* **2018**, *14*. doi:10.1145/3231598.
121. Kiess, J.; Guthier, B.; Kopf, S.; Effelsberg, W. SeamCrop: Changing the Size and Aspect Ratio of Videos. Proceedings of the 4th Workshop on Mobile Video; Association for Computing Machinery: New York, NY, USA, 2012; MoVid '12, p. 13–18. doi:10.1145/2151677.2151681.
122. Li, Y.; Tian, Y.; Yang, J.; Duan, L.Y.; Gao, W. Video Retargeting with Multi-Scale Trajectory Optimization. Proceedings of the International Conference on Multimedia Information Retrieval; Association for Computing Machinery: New York, NY, USA, 2010; MIR '10, p. 45–54. doi:10.1145/1743384.1743399.
123. Liu, D.; Wu, Z.; Lin, X.; Ji, R. Towards perceptual video cropping with curve fitting. *Multimedia Tools and Applications* **2014**, *75*, 12465–12475.
124. Li, B.; Duan, L.Y.; Wang, J.; Ji, R.; Lin, C.W.; Gao, W. Spatiotemporal Grid Flow for Video Retargeting. *IEEE Transactions on Image Processing* **2014**, *23*, 1615–1628. doi:10.1109/TIP.2014.2305843.
125. Wang, Y.S.; Hsiao, J.H.; Sorkine, O.; Lee, T.Y. Scalable and Coherent Video Resizing with Per-Frame Optimization. *ACM Trans. Graph.* **2011**, *30*. doi:10.1145/2010324.1964983.

126. Wang, Y.S.; Fu, H.; Sorkine, O.; Lee, T.Y.; Seidel, H.P. Motion-Aware Temporal Coherence for Video Resizing. *ACM Trans. Graph.* **2009**, *28*, 1–10. doi:10.1145/1618452.1618473.
127. Zwicker, M.; Pfister, H.; van Baar, J.; Gross, M. EWA splatting. *IEEE Transactions on Visualization and Computer Graphics* **2002**, *8*, 223–238. doi:10.1109/TVCG.2002.1021576.
128. Zhang, J.; Li, S.; Kuo, C.C.J. Compressed-Domain Video Retargeting. *IEEE Transactions on Image Processing* **2014**, *23*, 797–809. doi:10.1109/TIP.2013.2294541.
129. Yan, B.; Yuan, B.; Yang, B. Effective Video Retargeting With Jittery Assessment. *IEEE Transactions on Multimedia* **2014**, *16*, 272–277. doi:10.1109/TMM.2013.2286112.
130. Wolf, L.; Guttmann, M.; Cohen-Or, D. Non-homogeneous Content-driven Video-retargeting. 2007 IEEE 11th International Conference on Computer Vision, 2007, pp. 1–6. doi:10.1109/ICCV.2007.4409010.
131. Wang, Y.S.; Lin, H.C.; Sorkine, O.; Lee, T.Y. Motion-Based Video Retargeting with Optimized Crop-and-Warp. *ACM Trans. Graph.* **2010**, *29*. doi:10.1145/1778765.1778827.
132. Nie, Y.; Zhang, Q.; fang Wang, R.; Xiao, C. Video retargeting combining warping and summarizing optimization. *The Visual Computer* **2013**, *29*, 785–794.
133. Zhang, Y.F.; Hu, S.; Martin, R.R. Shrinkability Maps for Content-Aware Video Resizing. *Computer Graphics Forum* **2008**, *27*.
134. Liu, Y.; Sun, L.; Yang, S. A retargeting method for stereoscopic 3D video. *Computational Visual Media* **2015**, *1*, 119–127.
135. Shao, F.; Lin, W.; Fu, R.; Yu, M.; Jiang, G. Optimizing multiview video plus depth retargeting technique for stereoscopic 3D displays. *Opt. Express* **2017**, *25*, 12478–12492. doi:10.1364/OE.25.012478.
136. Rubinstein, M.; Shamir, A.; Avidan, S. Improved Seam Carving for Video Retargeting. *ACM Trans. Graph.* **2008**, *27*, 1–9. doi:10.1145/1360612.1360615.
137. Avidan, S.; Shamir, A. Seam Carving for Content-Aware Image Resizing. ACM SIGGRAPH 2007 Papers; Association for Computing Machinery: New York, NY, USA, 2007; SIGGRAPH '07, p. 10–es. doi:10.1145/1275808.1276390.
138. Sun, D. Volumetric Seam Carving. PhD thesis, 2017. Copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2021-08-03.
139. Kaur, H.; Kour, S.; Sen, D. Video retargeting through spatio-temporal seam carving using Kalman filter. *IET Image Process.* **2019**, *13*, 1862–1871.
140. Hsin, H.C. Video retargeting based on SH equalisation and seam carving. *IET Image Processing* **2019**, *13*, 1333–1340, [https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/iet-ipr.2018.5192]. doi:https://doi.org/10.1049/iet-ipr.2018.5192.
141. Furuta, R.; Tsubaki, I.; Yamasaki, T. Fast Volume Seam Carving With Multipass Dynamic Programming. *IEEE Transactions on Circuits and Systems for Video Technology* **2018**, *28*, 1087–1101. doi:10.1109/TCSVT.2016.2620563.
142. Nguyen, H.T.; Won, C.S. Stereo Video Retargeting with Representative Seams in a Group of Stereoscopic Frames. *ETRI Journal* **2013**, *35*, 980–989, [https://onlinelibrary.wiley.com/doi/pdf/10.4218/etrij.13.2013.0032]. doi:https://doi.org/10.4218/etrij.13.2013.0032.
143. Zhang, K.; Chao, W.L.; Sha, F.; Grauman, K. Video Summarization with Long Short-term Memory, 2016, [arXiv:cs.CV/1605.08110].
144. Zhao, B.; Xing, E.P. Quasi Real-Time Summarization for Consumer Videos. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2513–2520. doi:10.1109/CVPR.2014.322.
145. Poleg, Y.; Arora, C.; Peleg, S. Temporal Segmentation of Egocentric Videos. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2537–2544. doi:10.1109/CVPR.2014.325.
146. Rasheed, Z.; Shah, M. Scene detection in Hollywood movies and TV shows. 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings., 2003, Vol. 2, pp. II–343. doi:10.1109/CVPR.2003.1211489.
147. Potapov, D.; Douze, M.; Harchaoui, Z.; Schmid, C. Category-Specific Video Summarization. Computer Vision – ECCV 2014; Fleet, D.; Pajdla, T.; Schiele, B.; Tuytelaars, T., Eds.; Springer International Publishing: Cham, 2014; pp. 540–555.
148. Amel, A.M.; Abdelali, A.B.; Mtibaa, A. Video shot boundary detection using motion activity descriptor. *CoRR* **2010**, *abs/1004.4605*, [1004.4605].
149. Luo, J.; Papin, C.; Costello, K. Towards Extracting Semantically Meaningful Key Frames From Personal Video Clips: From Humans to Computers. *IEEE Transactions on Circuits and Systems for Video Technology* **2009**, *19*, 289–301. doi:10.1109/TCSVT.2008.2009241.
150. Zhou, F.; Kang, S.B.; Cohen, M.F. Time-Mapping Using Space-Time Saliency. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 3358–3365. doi:10.1109/CVPR.2014.429.
151. Bengio, S.; Pereira, F.; Singer, Y.; Strelow, D. Group Sparse Coding. Advances in Neural Information Processing Systems; Bengio, Y.; Schuurmans, D.; Lafferty, J.; Williams, C.; Culotta, A., Eds. Curran Associates, Inc., 2009, Vol. 22.
152. Li, J.; Yao, T.; Ling, Q.; Mei, T. Detecting shot boundary with sparse coding for video summarization. *Neurocomputing* **2017**, *266*, 66–78.
153. Sreeja, M.; Kovoor, B.C. Towards genre-specific frameworks for video summarisation: A survey. *Journal of Visual Communication and Image Representation* **2019**, *62*, 340–358. doi:https://doi.org/10.1016/j.jvcir.2019.06.004.
154. Li, P.; Guo, Y.; Sun, H. Multi-keyframe abstraction from videos. 2011 18th IEEE International Conference on Image Processing, 2011, pp. 2473–2476. doi:10.1109/ICIP.2011.6116162.
155. Goldman, D.B.; Curless, B.; Salesin, D.; Seitz, S.M. Schematic Storyboarding for Video Visualization and Editing. *ACM Trans. Graph.* **2006**, *25*, 862–871. doi:10.1145/1141911.1141967.
156. Truong, B.T.; Venkatesh, S. Video Abstraction: A Systematic Review and Classification. *ACM Trans. Multimedia Comput. Commun. Appl.* **2007**, *3*, 3–es. doi:10.1145/1198302.1198305.

157. Gong, B.; Chao, W.L.; Grauman, K.; Sha, F. Diverse Sequential Subset Selection for Supervised Video Summarization. Advances in Neural Information Processing Systems; Ghahramani, Z.; Welling, M.; Cortes, C.; Lawrence, N.; Weinberger, K.Q., Eds. Curran Associates, Inc., 2014, Vol. 27.
158. Mann, S. 'WearCam' (The wearable camera): personal imaging systems for long-term use in wearable tetherless computer-mediated reality and personal photo/videographic memory prosthesis. Digest of Papers. Second International Symposium on Wearable Computers (Cat. No.98EX215), 1998, pp. 124–131. doi:10.1109/ISWC.1998.729538.
159. Ng, H.W.; Sawahata, Y.; Aizawa, K. Summarization of wearable videos using support vector machine. Proceedings. IEEE International Conference on Multimedia and Expo, 2002, Vol. 1, pp. 325–328 vol.1. doi:10.1109/ICME.2002.1035784.
160. Ghosh, J. Discovering Important People and Objects for Egocentric Video Summarization. Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); IEEE Computer Society: USA, 2012; CVPR '12, p. 1346–1353.
161. Khosla, A.; Hamid, R.; Lin, C.J.; Sundaresan, N. Large-Scale Video Summarization Using Web-Image Priors. 2013 IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2698–2705. doi:10.1109/CVPR.2013.348.
162. Gygli, M.; Grabner, H.; Riemenschneider, H.; Van Gool, L. Creating Summaries from User Videos. Computer Vision – ECCV 2014; Fleet, D.; Pajdla, T.; Schiele, B.; Tuytelaars, T., Eds.; Springer International Publishing: Cham, 2014; pp. 505–520.
163. Gygli, M.; Grabner, H.; Riemenschneider, H.; Nater, F.; Gool, L.V. The Interestingness of Images. 2013 IEEE International Conference on Computer Vision, 2013, pp. 1633–1640. doi:10.1109/ICCV.2013.205.
164. Gygli, M.; Grabner, H.; Van Gool, L. Video summarization by learning submodular mixtures of objectives. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3090–3098. doi:10.1109/CVPR.2015.7298928.
165. Poleg, Y.; Halperin, T.; Arora, C.; Peleg, S. EgoSampling: Fast-forward and stereo for egocentric videos. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 4768–4776. doi:10.1109/CVPR.2015.7299109.
166. Sahu, A.; Chowdhury, A.S. First person video summarization using different graph representations. *Pattern Recognition Letters* **2021**, *146*, 185–192. doi:https://doi.org/10.1016/j.patrec.2021.03.013.
167. Xiong, B.; Grauman, K. Detecting Snap Points in Egocentric Video with a Web Photo Prior. 2014, pp. 282–298. doi:10.1007/978-3-319-10602-1_19.
168. del Molino, A.G.; Tan, C.; Lim, J.H.; Tan, A.H. Summarization of Egocentric Videos: A Comprehensive Survey. *IEEE Transactions on Human-Machine Systems* **2017**, *47*, 65–76. doi:10.1109/THMS.2016.2623480.
169. Zhou, P.; Xu, T.; Yin, Z.; Liu, D.; Chen, E.; Lv, G.; Li, C. Character-Oriented Video Summarization With Visual and Textual Cues. *IEEE Transactions on Multimedia* **2020**, *22*, 2684–2697. doi:10.1109/TMM.2019.2960594.
170. Mindek, P.; Čmolík, L.; Viola, I.; Gröller, E.; Bruckner, S. Automatized Summarization of Multiplayer Games. Proceedings of the 31st Spring Conference on Computer Graphics; Association for Computing Machinery: New York, NY, USA, 2015; SCCG '15, p. 73–80. doi:10.1145/2788539.2788549.
171. Hussain, T.; Muhammad, K.; Ding, W.; Lloret, J.; Baik, S.W.; de Albuquerque, V.H.C. A comprehensive survey of multi-view video summarization. *Pattern Recognition* **2021**, *109*, 107567. doi:https://doi.org/10.1016/j.patcog.2020.107567.
172. Meng, J.; Wang, S.; Wang, H.; Yuan, J.; Tan, Y.P. Video Summarization Via Multiview Representative Selection. *IEEE Transactions on Image Processing* **2018**, *27*, 2134–2145. doi:10.1109/TIP.2017.2789332.
173. De Leo, C.; Manjunath, B.S. Multicamera Video Summarization from Optimal Reconstruction; Springer-Verlag: Berlin, Heidelberg, 2010; ACCV'10, p. 94–103.
174. Kuanar, S.K.; Ranga, K.B.; Chowdhury, A.S. Multi-View Video Summarization Using Bipartite Matching Constrained Optimum-Path Forest Clustering. *IEEE Transactions on Multimedia* **2015**, *17*, 1166–1173. doi:10.1109/TMM.2015.2443558.
175. Wang, L.; Fang, X.; Guo, Y.; Fu, Y. Multi-view Metric Learning for Multi-view Video Summarization. 2016 International Conference on Cyberworlds (CW), 2016, pp. 179–182. doi:10.1109/CW.2016.38.
176. Panda, R.; Das, A.; Roy-Chowdhury, A.K. Embedded sparse coding for summarizing multi-view videos. 2016 IEEE International Conference on Image Processing (ICIP), 2016, pp. 191–195. doi:10.1109/ICIP.2016.7532345.
177. Ji, Z.; Zhang, Y.; Pang, Y.; Li, X. Hypergraph dominant set based multi-video summarization. *Signal Processing* **2018**, *148*, 114–123. doi:https://doi.org/10.1016/j.sigpro.2018.01.028.
178. Irani, M.; Anandan, P. Video indexing based on mosaic representations. *Proceedings of the IEEE* **1998**, *86*, 905–921. doi:10.1109/5.664279.
179. Rav-Acha, A.; Pritch, Y.; Peleg, S. Making a Long Video Short: Dynamic Video Synopsis. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* **2006**, *1*, 435–441.
180. Kirkpatrick, S.; Gelatt, C.D.; Vecchi, M.P. Optimization by Simulated Annealing. *Science* **1983**, *220*, 671–680, [https://www.science.org/doi/p doi:10.1126/science.220.4598.671.
181. Pritch, Y.; Rav-Acha, A.; Gutman, A.; Peleg, S. Webcam Synopsis: Peeking Around the World. 2007 IEEE 11th International Conference on Computer Vision, 2007, pp. 1–8. doi:10.1109/ICCV.2007.4408934.
182. Baskurt, K.B.; Samet, R. Video synopsis: A survey. *Computer Vision and Image Understanding* **2019**, *181*, 26–38. doi:https://doi.org/10.1016/j.c
183. Mahapatra, A.; Sa, P.K.; Majhi, B. A multi-view video synopsis framework. 2015 IEEE International Conference on Image Processing (ICIP), 2015, pp. 1260–1264. doi:10.1109/ICIP.2015.7351002.
184. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*, 2 ed.; Cambridge University Press: USA, 2003.
185. Mahapatra, A.; Mishra, T.K.; Sa, P.K.; Majhi, B. Human recognition system for outdoor videos using Hidden Markov model. *AEU - International Journal of Electronics and Communications* **2014**, *68*, 227–236. doi:https://doi.org/10.1016/j.aeue.2013.08.011.

186. Rakotomamonjy, A.; Bach, F.R.; Canu, S.; Grandvalet, Y. SimpleMKL. *Journal of Machine Learning Research* **2008**, *9*, 2491–2521.
187. Rav-Acha, A.; Pritch, Y.; Lischinski, D.; Peleg, S. Dynamosaics: video mosaics with non-chronological time. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005, Vol. 1, pp. 58–65 vol. 1. doi:10.1109/CVPR.2005.137.
188. Liu, F.; Hu, Y.h.; Gleicher, M.L. Discovering Panoramas in Web Videos. Proceedings of the 16th ACM International Conference on Multimedia; Association for Computing Machinery: New York, NY, USA, 2008; MM '08, p. 329–338. doi:10.1145/1459359.1459404.
189. Szeliski, R. Image Alignment and Stitching: A Tutorial. *Found. Trends. Comput. Graph. Vis.* **2006**, *2*, 1–104. doi:10.1561/0600000009.
190. Bazin, J.C.; (Kuster), C.P.; Yu, G.; Martin, T.; Jacobson, A.; Gross, M. Physically Based Video Editing. *Computer Graphics Forum* **2016**, *35*, 421–429, [https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13039]. doi:https://doi.org/10.1111/cgf.13039.
191. Bai, J.; Agarwala, A.; Agrawala, M.; Ramamoorthi, R. Selectively De-Animating Video. *ACM Trans. Graph.* **2012**, *31*. doi:10.1145/2185520.2185562.
192. Lucas, B.D.; Kanade, T. An Iterative Image Registration Technique with an Application to Stereo Vision. Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 1981; IJCAI'81, p. 674–679.
193. Zhang, H.; Sciutto, C.; Agrawala, M.; Fatahalian, K. Vid2Player: Controllable Video Sprites that Behave and Appear like Professional Tennis Players. *ArXiv* **2020**, *abs/2008.04524*.
194. Chang, C.S.; Chu, H.K.; Mitra, N.J. Interactive Videos: Plausible Video Editing using Sparse Structure Points. *Computer Graphics Forum* **2016**, *35*, 489–500, [https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12849]. doi:https://doi.org/10.1111/cgf.12849.
195. Pollefeys, M.; Gool, L.V.; Vergauwen, M.; Verbiest, F.; Cornelis, K.; Tops, J.; Koch, R. Visual Modeling with a Hand-Held Camera. *International Journal of Computer Vision* **2004**, *59*, 207–232.
196. Kasten, Y.; Ofri, D.; Wang, O.; Dekel, T. Layered Neural Atlases for Consistent Video Editing. *ACM Trans. Graph.* **2021**, *40*. doi:10.1145/3478513.3480546.
197. Fried, O.; Agrawala, M. Puppet Dubbing, 2019, [arXiv:cs.GR/1902.04285].
198. Lu, Z.; Ling, Q.; Li, H.; Li, W. Video restoration based on a novel second order nonlocal total variation model. *Signal Processing* **2017**, *133*, 79–96. doi:https://doi.org/10.1016/j.sigpro.2016.10.009.
199. Li, M.; Liu, J.; Sun, X.; Xiong, Z. Image/Video Restoration via Multiplanar Autoregressive Model and Low-Rank Optimization. *ACM Trans. Multimedia Comput. Commun. Appl.* **2019**, *15*. doi:10.1145/3341728.
200. Bai, T.; Wang, B.; Nguyen, D.; Jiang, S. Deep dose plugin: towards real-time Monte Carlo dose calculation through a deep learning-based denoising algorithm. *Machine Learning : Science and Technology* **2021**, *2*. Copyright - © 2021. This work is published under http://creativecommons.org/licenses/by/4.0 (the "License"). Notwithstanding the ProQuest Terms and Conditions, you may use this content in accordance with the terms of the License; Last updated - 2021-04-20.
201. Söderkvist, O. Computer Vision Classification of Leaves from Swedish Trees. 2001.
202. Quattoni, A.; Torralba, A. Recognizing indoor scenes. 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 413–420. doi:10.1109/CVPR.2009.5206537.
203. Anwar, S.; Tahir, M.; Li, C.; Mian, A.; Khan, F.S.; Muzaffar, A.W. Image Colorization: A Survey and Dataset, 2020, [arXiv:cs.CV/2008.10774].
204. Xiao, J.; Hays, J.; Ehinger, K.A.; Oliva, A.; Torralba, A. SUN database: Large-scale scene recognition from abbey to zoo. 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010, pp. 3485–3492. doi:10.1109/CVPR.2010.5539970.
205. Bossard, L.; Guillaumin, M.; Van Gool, L. Food-101 – Mining Discriminative Components with Random Forests. Computer Vision – ECCV 2014; Fleet, D.; Pajdla, T.; Schiele, B.; Tuytelaars, T., Eds.; Springer International Publishing: Cham, 2014; pp. 446–461.
206. Xiao, J.; Ehinger, K.A.; Oliva, A.; Torralba, A. Recognizing scene viewpoint using panoramic place representation. 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 2695–2702. doi:10.1109/CVPR.2012.6247991.
207. Li, L.J.; Fei-Fei, L. What, where and who? Classifying events by scene and object recognition. 2007 IEEE 11th International Conference on Computer Vision, 2007, pp. 1–8. doi:10.1109/ICCV.2007.4408872.
208. Scherer, S.; Layher, G.; Kane, J.; Neumann, H.; Campbell, N. An audiovisual political speech analysis incorporating eye-tracking and perception data. Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12); European Language Resources Association (ELRA): Istanbul, Turkey, 2012; pp. 1114–1120.
209. Liu, Y.; Qiao, M.; Xu, M.; Li, B.; Hu, W.; Borji, A. Learning to Predict Salient Faces: A Novel Visual-Audio Saliency Model. *ArXiv* **2020**, *abs/2103.15438*.
210. Yang, B.; Nevatia, R. Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1918–1925. doi:10.1109/CVPR.2012.6247892.
211. Thrun, S.; Fox, D.; Burgard, W.; Dellaert, F. Robust Monte Carlo localization for mobile robots. *Artificial Intelligence* **2001**, *128*, 99–141. doi:https://doi.org/10.1016/S0004-3702(01)00069-8.
212. Loui, A.C.; Luo, J.; Chang, S.F.; Ellis, D.P.W.; Jiang, W.; Kennedy, L.S.; Lee, K.; Yanagawa, A. Kodak's consumer video benchmark data set: concept definition and annotation. MIR '07, 2007.

213. Fransens, R.; Strecha, C.; Van Gool, L. Parametric Stereo for Multi-pose Face Recognition and 3D-Face Modeling. Analysis and Modelling of Faces and Gestures; Zhao, W.; Gong, S.; Tang, X., Eds.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2005; pp. 109–124.
214. Hodosh, M.; Young, P.; Rashtchian, C.; Hockenmaier, J. Cross-Caption Coreference Resolution for Automatic Image Understanding. Proceedings of the Fourteenth Conference on Computational Natural Language Learning; Association for Computational Linguistics: Uppsala, Sweden, 2010; pp. 162–171.
215. Lazebnik, S.; Schmid, C.; Ponce, J. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), 2006, Vol. 2, pp. 2169–2178. doi:10.1109/CVPR.2006.68.
216. Liu, Y.J.; Luo, X.; Xuan, Y.M.; Chen, W.F.; Fu, X.L. Image Retargeting Quality Assessment. *Computer Graphics Forum* **2011**. doi:10.1111/j.1467-8659.2011.01881.x.
217. Tewari, A.; Fried, O.; Thies, J.; Sitzmann, V.; Lombardi, S.; Sunkavalli, K.; Martin-Brualla, R.; Simon, T.; Saragih, J.; Nießner, M.; Pandey, R.; Fanello, S.; Wetzstein, G.; Zhu, J.Y.; Theobalt, C.; Agrawala, M.; Shechtman, E.; Goldman, D.B.; Zollhöfer, M. State of the Art on Neural Rendering, 2020, [arXiv:cs.CV/2004.03805].
218. Dong, C.; Loy, C.C.; He, K.; Tang, X. Image Super-Resolution Using Deep Convolutional Networks, 2015, [arXiv:cs.CV/1501.00092].
219. Wang, Z.; Chen, J.; Hoi, S.H. Deep Learning for Image Super-Resolution: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2021**, *43*, 3365–3387. doi:10.1109/TPAMI.2020.2982166.
220. Anwar, S.; Khan, S.; Barnes, N. A Deep Journey into Super-resolution: A survey, 2020, [arXiv:cs.CV/1904.07523].
221. Yuzhi, Z.; Lai-Man, P.; Xuehui, W.; Kangcheng, L.; Yujia, Z.; Wing-Yin, Y.; Pengfei, X.; Jingjing, X. Legacy Photo Editing with Learned Noise Prior, 2020, [arXiv:cs.CV/2011.11309].
222. Wu, C.; Gao, T. Image Denoise Methods Based on Deep Learning. *Journal of Physics: Conference Series* **2021**, *1883*. Copyright - © 2021. This work is published under http://creativecommons.org/licenses/by/3.0/ (the "License"). Notwithstanding the ProQuest Terms and Conditions, you may use this content in accordance with the terms of the License; Last updated - 2021-06-21.
223. Arar, M.; Danon, D.; Cohen-Or, D.; Shamir, A. Image Resizing by Reconstruction from Deep Features, 2021, [arXiv:cs.CV/1904.08475].
224. Dekel Basha, T.; Moses, Y.; Avidan, S. Stereo Seam Carving a Geometrically Consistent Approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2013**, *35*, 2513–2525. doi:10.1109/TPAMI.2013.46.
225. Lu, D.; Ma, H.; Liu, L. Visually preserving stereoscopic image retargeting using depth carving. *Journal of Electronic Imaging* **2016**, *25*, 1 – 12. doi:10.1117/1.JEI.25.2.023029.
226. Su, J.W.; Chu, H.K.; Huang, J.B. Instance-aware Image Colorization, 2020, [arXiv:cs.CV/2005.10825].
227. Cheng, Z.; Yang, Q.; Sheng, B. Deep Colorization, 2016, [arXiv:cs.CV/1605.00075].
228. Carlucci, F.M.; Russo, P.; Caputo, B. $(DE)^2CO$: Deep Depth Colorization. *IEEE Robotics and Automation Letters* **2018**, *3*, 2386–2393. doi:10.1109/LRA.2018.2812225.
229. Zhang, R.; Isola, P.; Efros, A.A. Colorful Image Colorization, 2016, [arXiv:cs.CV/1603.08511].
230. Zhang, R.; Zhu, J.Y.; Isola, P.; Geng, X.; Lin, A.S.; Yu, T.; Efros, A.A. Real-Time User-Guided Image Colorization with Learned Deep Priors, 2017, [arXiv:cs.CV/1705.02999].
231. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN, 2018, [arXiv:cs.CV/1703.06870].
232. Žeger, I.; Grgic, S.; Vuković, J.; Šišul, G. Grayscale Image Colorization Methods: Overview and Evaluation. *IEEE Access* **2021**, *9*, 113326–113346. doi:10.1109/ACCESS.2021.3104515.
233. Jiang, Y.; Jia, X.; Zhang, L.; Yuan, Y.; Chen, L.; Yin, G. Image-to-Image Style Transfer Based on the Ghost Module. *Computers, Materials, and Continua* **2021**, *68*, 4051–4067. Copyright - © 2021. This work is licensed under https://creativecommons.org/licenses/by/4.0/ (the "License"). Notwithstanding the ProQuest Terms and Conditions, you may use this content in accordance with the terms of the License; Last updated - 2021-06-02.
234. Hicsonmez, S.; Samet, N.; Akbas, E.; Sahin, P.D. GANILLA: Generative Adversarial Networks for Image to Illustration Translation. *Image Vis. Comput.* **2020**, *95*, 103886.
235. Wang, J.; Agrawala, M.; Cohen, M.F. Soft Scissors: An Interactive Tool for Realtime High Quality Matting. *ACM Trans. Graph.* **2007**, *26*, 9–es. doi:10.1145/1276377.1276389.
236. Kim, G.; Xing, E.P. Jointly Aligning and Segmenting Multiple Web Photo Streams for the Inference of Collective Photo Storylines. 2013 IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 620–627. doi:10.1109/CVPR.2013.86.
237. Grabler, F.; Agrawala, M.; Li, W.; Dontcheva, M.; Igarashi, T. Generating Photo Manipulation Tutorials by Demonstration. *ACM Trans. Graph.* **2009**, *28*. doi:10.1145/1531326.1531372.
238. Ramanathan, V.; Joulin, A.; Liang, P.; Fei-Fei, L. Linking People in Videos with "Their" Names Using Coreference Resolution. Computer Vision – ECCV 2014; Fleet, D.; Pajdla, T.; Schiele, B.; Tuytelaars, T., Eds.; Springer International Publishing: Cham, 2014; pp. 95–110.
239. Haurilet, M.L.; Tapaswi, M.; Al-Halah, Z.; Stiefelhagen, R. Naming TV characters by watching and analyzing dialogs. 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), 2016, pp. 1–9. doi:10.1109/WACV.2016.7477560.
240. Nagrani, A.; Zisserman, A. From Benedict Cumberbatch to Sherlock Holmes: Character Identification in TV series without a Script, 2018, [arXiv:cs.CV/1801.10442].
241. Alayrac, J.B.; Sivic, J.; Laptev, I.; Lacoste-Julien, S. Joint Discovery of Object States and Manipulation Actions, 2017, [arXiv:cs.CV/1702.02738].

242. Huang, D.A.; Lim, J.J.; Fei-Fei, L.; Niebles, J.C. Unsupervised Visual-Linguistic Reference Resolution in Instructional Videos. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* **2017**, pp. 1032–1041.
243. Huang*, D.A.; Buch*, S.; Dery, L.; Garg, A.; Fei-Fei, L.; Niebles, J.C. Finding "It": Weakly-Supervised, Reference-Aware Visual Grounding in Instructional Videos. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
244. Xia, J.; Rao, A.; Huang, Q.; Xu, L.; Wen, J.; Lin, D. Online Multi-modal Person Search in Videos, 2020, [arXiv:cs.CV/2008.03546].
245. Huang, Q.; Liu, W.; Lin, D. Person Search in Videos with One Portrait Through Visual and Temporal Links, 2018, [arXiv:cs.CV/1807.10510].
246. Li, D.; Xu, T.; Zhou, P.; He, W.; Hao, Y.; Zheng, Y.; Chen, E. Social Context-Aware Person Search in Videos via Multi-Modal Cues. *ACM Trans. Inf. Syst.* **2021**, *40*. doi:10.1145/3480967.
247. Rudoy, D.; Goldman, D.B.; Shechtman, E.; Zelnik-Manor, L. Learning Video Saliency from Human Gaze Using Candidate Selection. 2013 IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1147–1154. doi:10.1109/CVPR.2013.152.