Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.777744

DEEP OCCLUSION-BASED KEY BACKGROUND EXTRACTION USING POINTREND INSTANCE SEGMENTATION

Suresha M*,[1][0000-0003-0668-926X], Kuppa S[1,2][0000-0002-7897-6789] and Raghu Kumar D S[1,3][0000-0002-0120-4430]

[1] Kuvempu University, Karnataka 577451, INDIA
[1]suresham@kuvempu.ac.in,[2]kuppa1993@gmail.com,[3]rg12.clk@gmail.com

Abstract

Multimedia information retrieval, key-frame extraction is a common need-based problem-specific information retrieval technique in video content. This information retrieval technique reduces the computational power by eliminating a huge set of unwanted frames through the removal of unnecessary content and retrieves key feature frames based on specific video analysis problems. In video analysis classification and captioning of video content using deep neural models, spatio-temporal clues are the most important features. In these two features, modeling of spatial feature is an object understanding by its spatial background, which requires a huge set of frames because objects are occluded by temporal, moving objects. Our instance-based key-frame extraction techniques solve this problem by extracting clear background frames in a video sequence. Here we use a popular PointRend instance segmentation approach to find the moving instances in a video sequence. Effective Instance Prediction Matrix (EIPM) computes predicted instances, which are then evaluated to find the least minimum instance occluded key-frame. We discuss the subdivision techniques of PointRend instance segmentation and results on video datasets. We present results on instance-based key-frame extraction with different datasets and neural models.

Keywords: Instance Segmentation, Occlusions, Deep Learning, Background Extraction, Key Background

1. Introduction

Video analysis is an actively studied research area in multimedia domain because a considerable amount of data spread over in big data is a video data. Analysis of raw video data creates many research avenues to solve real-world problems. In research, it deals with object detection[29], classification[38], and captioning[39] or description generation. This could be possible by modelling video data with suitable feature engineering techniques. The feature modelling takes to extract abundant hidden features in raw video data. The features modelling techniques are classified into two categories based on their extraction techniques such as classical and modern machine vision[33] techniques. For example, in a classical vision, features are HOG[28], HOF[36], STIP[21] and Dense trajectory[35], in modern machine vision

optical flow feature evaluation using flow nets[16] and spatio-temporal feature modelling using hybrid deep learning[38] frameworks.

Machine Vision feature extraction models are given more prominent results compared to classical vision[33] techniques. Because here a considerable amount of raw data is used to represent the modelling because all such modern machine vision models take representation learning. This can need minimum feature engineering[22] techniques compared to classical vision. Therefore, recent multimedia researches focused on these techniques with a huge amount of raw data and computational power. However, some extinct cases can face some critical problem by using an improper alignment of raw data; this can take a huge amount of computational power.

Minimization of computational power and improving model accuracy is one of the important tasks in deep learning[33]. Therefore preprocessing raw data, modelling the abundant features effectively, hyperactive tuning the parameter and selective allocation[17] of feature maps actively engaged areas to optimize the deep neural networks. Recently pre-processing or analysis of raw data aimed to be problem-centric to train deep neural architecture. Different types of pre-processing techniques used to extract valuable information in raw video data which improves the computational efficiency, for example background extraction[9], key action recognition[3], attention generation[40], and salient activity[4], or object detection[31] and key frame extraction[1].

Video data is made up of a set of frames that can include both required and unwanted data frames. Depending on the issue, we can divide data frames into necessary and optional categories. For example, in order to detect an object, a clear background is needed; some action recognition problems, key action frames are required; and in some other situations, such as captioning or description generation[32], all frames in a video data are required. In general, moving and static objects considered as core features. Mapping of these features have faced many problems without knowing the background information clearly, because in real-world scenario background is occluded by moving instances or moving objects. This problem could be solved by the extraction of a clear background extraction framework.

Background extraction improves the rate of object detection accuracy by clear identification of objects without moving instance occlusion. In computer vision, there are several approaches designed to solve this problem. Here discussed a new way to extract a key background frame using state-of-the-art deep learning approaches. The emergence of deep learning and its pre-trained models leads to good accuracy. Using such deep learning models for object segmentation gives significant results in computer vision. Segmentation is an important task to look at every object, segment each instance based on its pixel-level or object of interest, which helps to classify contents in video frames. Generally, segmentation is focused on dividing images into set of pixels[19] and each set is derived based on similarity in texture, colour or use of some specific requirements. There are different types of segmentations available in the real-world such as schematic segmentation[11] , instance segmentation [19] etc. Use instance

segmentation instead of semantic segmentation, even though it does not categories each pixel in an image. It primarily focuses on object interest and creates a mask around every object in an image instead of bounding boxes. A pre-trained mask-RCNN [14] network performs instance segmentation of each frame. It shows generalized tasks like detection of objects, human pose segments in every instance with precision. Compared to semantic segmentation, instance segmentation[19] gives good results but it is difficult to comprehend the object boundaries in complex objects and events.

Effective identification of key-frame in complex and crisply occluded instances without losing object boundaries preserve the background image features. In a densely occluded frame, segmentation of instances using instance segmentation oversampling the image features results in multiple instances considered as single. As a result, a unique perspective of image segmentation is required to solve the rendering problem by analogizing some classical computer graphics methods for efficient rendering with over and under-sampling challenges encountered in pixel-level labelling tasks. PointRend [19] efficiently detects the objects and preserves the output resolutions. PointRend produces crisp object boundaries.

After successful detection of every object in a complex moving instance on a static background, each segmented objects with respective frames are masked and grouped. This can be configured by batch, each batch contains a sequence of frames, number batches are created dynamically based on equal number of frames in a video. In each batch, a successfully detected instance is organized into a dynamic array called Effective Instance Prediction Matrix (EPIM). EPIM organizes each batch with respective frame numbers in rows and columns. The detected instances are inserted into columns with only the least minimum number instance on every frame the comparison and insertion algorithm find the least minimum element and insert the same respectively. Using EPIM choose the best minimum instance occluded frame as a keyframe.

Our approach is to provide the best way to find minimum moving instance frames in a real-time video sequence. This is novel way to find the background frame in real-world scenarios instead of finding background subtraction methods and without knowing the ground truth of the video content. This method improves computing accuracy in video analysis problems with effective usage of background objects without losing abstract features. Using this real-time key background frame extraction technique reduces the number of spatial frames in different vision problems also these selective feature frames are used for spatio-temporal feature modelling.

The remaining sections of the paper are organized in the following manner. The second section examines similar works. The proposed key frame extraction using PointRend instance segmentation is detailed in Section 3. Section 4 discusses the experimental findings and interpretation, and Section 5 discusses the conclusions and future work.

2. Related work

Instance Segmentation

In recent approaches [5, 24], Mask R- CNN models [14] took first place for instance segmentation methods. Regardless of object size, these region-based models typically masks on a 28X28 grid. This is adequate for diminutive objects, but it gives unfavorable performance by over-smoothing fine-level information in augmentative objects (as shown in Figure. 1, top-left). Bottom-up methods combine pixels to create object masks [2, 18, 23]. These techniques can achieve more comprehensive results. On most instance segmentation benchmarks, however, they lag behind region-based approaches [7, 8, 12]. Tensor Mask [6], an alternative sliding window strategy, predicts sharp high-resolution masks for large artefacts using a sophisticated network architecture. Tensor Mask [6] uses a sophisticated network architecture to generates masks with high-resolution for large artefacts, however, its accuracy is also slightly poor. In this paper, we demonstrate how a region-based segmentation model with PointRend [19] can generate masks over finer-level information while improve the accuracy of region oriented approaches.

Key Background Extraction

Key Background extraction is a problem-specific information retrieval technique that was used in [20] for salient behaviour recognition using semantic features based key frames retrieval, and in [25] for video summarization using multidimensional time series analysis and clustering techniques, and these methods were improved by integrating key-frames and feature fusion models[34]. Need-based information processing often eliminates unnecessary information based on the issue of precision, which leads to the reduction of dimensionality. Information reduction is a significant component for optimizing video analysis models. The spatial and temporal clues in video content are [32] core features of video classification and captioning; modelling of these spatio-temporal features required spatial and temporal frames; pooling of these frames required high computing power; reducing this computational requirement is a challenging task. Choosing the right spatial frame allows you to reduce computational requirements while also optimizing the model. As a result, various experiments have been carried out in order to find a cutting-edge key frame extraction method for optimizing models.

3. Outline of the proposed approach

The main area of this approach is aim to optimize the feature extraction algorithms in computer vision, effective analysis of video content and improve neural models with less number of layers etc. In this context reduction of input parameters with high impact features and select then allocation of channels reduce network complexity at the time of training are majorly served core areas. Preprocessing of the raw data is very essential before going to attend to these optimization problems to achieve a best result. Here key frames extraction is one of the techniques to reduce complexity in raw data and select suitable frames for state-of-art kind of problems. The key frames are extracted based on moving instances in each frame. Because we intend to find a clear background frame, using a popular instance segmentation model to identify or mask each moving instance in a video sequence. After successful identification of instances in a video frame, score the identified instances with Effective Instance Prediction Matrix (EIPM).

3.1 PointRend Instance Segmentation

The task of segmenting instances is to map pixels sampled on a regular grid to a label map or a set of label maps on same grid. Instance segmentation is mapping of the binary foreground vs background maps of each detected object. Modern machine vision tools such as CNN generated these maps. Due to minimal frequency sparse limits between object regions, the label maps created by these CNNs are almost smooth i.e the adjacent pixels are also categorized on the same label. The smooth areas would be excessively oversampled by a regular grid while undersampling object borders at the same time. This would result in computing smooth regions and blurred contours (as in Figure 1). This would be compromised between [26][14] oversampling and undersampling. As a result, models the rasterized image i.e. regular grid of pixels by renderer maps like 3D meshes using classical rendering techniques. The computation is not distributed evenly across the grid, despite the fact that the output is on a regular grid. Instead of computing pixel values by irregular subsets, dynamically select points over the image plane. As an example, quadtree-like sequencing pattern that successfully renders a high-resolution, anti-aliased image generated by the traditional subdivision [37] approach.

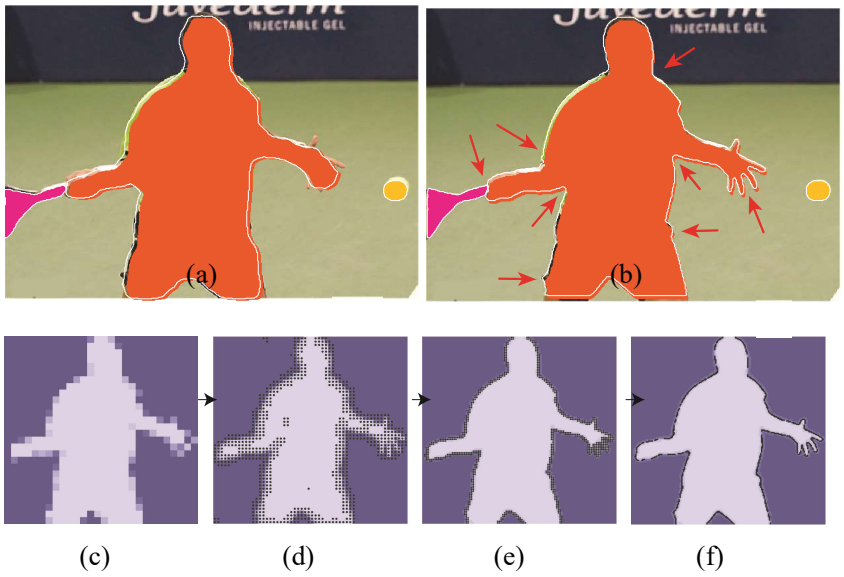


Figure 1: Example of oversampled (a, c & d) with normal smoothed (b, e & f)

The core principle of this PointRend instance segmentation is rendering problem and the application of classical ideas from computer graphics to the effective rendering of high-quality labelled maps (as in Figure. 1c and 1d). This can encapsulate by a deep neural (DNN) module called know as PointRend by using subdivision approach to fit a non-homogeneous collection of points in which labels to be computed. PointRend [19] module integrated into common neural architectures used for instance segmentation (example. Mask R-CNN[14]). Using an order of magnitude, the subdivision approach effectively computes high-resolution segmentation maps with less floating-points compare to direct dense computing. PointRend accepts a collection of typical segmentation feature mappings $f(x_i, y_i)$ are generated by CNN defined on the basis of regular grids also it produces quality of feature mappings $p(x'i, y' i)$ on

a fine grid. Instead of selection unwanted points on the resultant map, PointRend predicts useful points only. Interpolating *f* by using these predictions to extract point-wise features on selected useful points, and used a fine point header subnet to evaluate point-wise output labels (as in Figure. 2). Following that, we will discuss over the three important components of PointRend.

Figure 2: Example of Results of Mask RCNN (Left Image) *vs* Results of PointRend (Right Image)

Multiple CNN feature matrix over C channels $f \in \mathbb{R}^{C\times H\times W}$ are input to PointRend model, each specified vector on such a regular map (in image map generally 4 to 16 coarser), also performs estimation for the N target classes $p \in \mathbb{R}^{N\times H'\times W'}$ above a distinct (and probably a lot more) quality regular map, three main parts of PointRend:

(i) **A point chosen technique:** Predicts from a small number of real-value points while avoiding unnecessary computation for all pixels.

(ii) **For each chosen point:** A point-by-point function representation is obtained. Using bilinear insertion of *f* and minimum four nearest neighbors (NN) points over the regular maps of *f*, the features of a real-value point are computed.

(iii) **A point header**: A network trained to predict point-by-point representation of feature.

The PointRend model can be used for both semantic segmentation (for example, on FCNs [30]) and instance segmentation (for example, on Mask R-CNN [14]). PointRend is apply to every single region during segmentation process. Using coarse to fine predictions about a range of selected points it computed masks (as shown in Figure. 3). We will explain PointRend in terms

of instance segmentation because the entire image considered as a single region for semantic segmentation.

3.2 Instance-Based Key-Frame Extraction

Our method's state-of-the-art is the idea of predicting and selecting image frames based on a smaller number of moving instances captured in a video sequence. Intuitively, these frames should be selected more densely and nearly high-occluded regions, such as short-term to long-term motions without losing object boundaries. To develop this idea, we propose Effective Instance Prediction Matrix (EIPM). In each batch, select and elect operations helps to compare the number of predicted moving instances with previous consecutive frames. Based on these comparisons in EPIM only insert the minimum value of each frame index column other than updating it has some infinite values. Finally, a row contains a minimum value and infinite value set of frames in each batch. Using some sorting technique find less instance mapped frames.

Effective Instance Prediction Matrix

The core mechanism of the instance prediction matrix is the insertion of several instances followed by each batch concerning the frame index. To begin with, let P(B; F) denote the number of batches and frames in each batch in equation 1. The value of each frame index is Si and Ni, Si indicates an instance of the selected frame and Ni an index of the next selected frame. The value of Si as the selected least minimum value in each row and Ni is chosen if a predicted instance is greater than Si. After successful insertion of each frame index F, instances Si in every batch B then chose the least minimum instances in each row as key frame of the respective batch. The overall key frames of full video sequences taken by finding the least minimum by all batches.

$$EIPM(B;F)_i \in \mathbb{R}^o$$

$$:= \frac{1}{SI\ NI}\sum_{si,ni}^{n} \mathbb{E}_x[P(F,B) \propto \sim P(F,B_{-i})]:, si, ni \qquad \text{Eq.1}$$

Algorithm: 1 Select and Elect in EIPM
Input: X = (B, F, I, n), EIPM **(**B, F**)** $\in \mathbb{R}^{SI \times NI}$ Selection level γ > Ø

Initialize: B ≠ 0, F ≠ 0
for all 1≤ n **do**
for all 1 ≤ I ∈ B **do**
B←B'
F←F' { $\boldsymbol{IF\ argmin_{SI>NI \in F}\ then\ \|\ EIPM\ (B,F)_{SI}\ \|\ else\ \|\ EIPM(B,F)_{\infty}\ \|}$ }
I←I-1
end for
n = n-1
end for

Selective Prediction

Using PointRend Masks compute the number of instances and these are further compared with each least minimum value *Si* selected from the EPIM. *Algorithm 2* derived to find the least minimum value *Si* in EPIM. The new Instance value *Ni* is greater than the *Si*. The next *Ni* value will be updated as an infinite value. This can be derived by *Algorithm 1*, it proceeded by comparison between *Si* with next *Ni* if *Ni*<*Si* then the value of *Ni* inserted into the respective frame index in EPIM, otherwise index are updated with infinite values.

Algorithm: 2 Instance-Based Key Frame Selection in EPIM

Input: X = EIPM **(**B, F,K**)** $\in \mathbb{R}^{SI}$
Selection level $\gamma > Ø$

Initialize: $B \neq 0$, $F \neq 0$
for all $0 \leq I \in B$ **do**
B←B'
F←F' |{ $\boldsymbol{if\ argmin\ EIPM}[I]_{SI \in F}$ $\boldsymbol{then\ argmin\ EIPM}\ [I]_F$ }|
K←F
end if
I←I-1
end for

4. Experimental Setup and Results Analysis

4.1 Datasets Details. PointRend model have been trained by using the Cityscapes [8] and the COCO [7] benchmarks. In these datasets PointRend mask the fine-sharp objects boundaries, as shown in Fig. 2 (COCO and Cityscapes) and Fig. 8(Avenue Dataset). The has contain COCO has 80 categories that are annotated at the instance level. Cityscapes is a dataset of egocentric street scenes with 8 groups, 2975 training images, and 500 testing images. The images have a good resolution (1024 X 2048 pixels) than COCO and its pixel values is more accurate which helps to instance segmentations. Avenue[13] dataset is real-time metro station scenario dataset, which contain 37 videos with 30652 frames in total.

4.2 Architecture. The model designed based on multi-model perspective which contain Mask R-CNN along with a ResNet-50 [15] combined with Feature Pyramid Network (FPN) [30] backbone is used in for experimentation (as shown in Figure 4). Mask R-CNN mask head is a region oriented FCN that can disperse. We make the required changes for PointRend, as indicated by the "4x conv" notation.

4.2.1 Prediction of coarse mask. To determine the coarse, it replaces the 4x conv mask that parallels Mask R- CNN, extracting a 14 x 14 feature box and producing a 77% masks for each bounding boxes. Specifically, used bilinear interpolation to derive a 14x14 function which maps P2 level by the FPN for each bounding box. Inside the bounding box, the features are calculated on a standard grid (this process can be thought of as a simplified version with 256 output size, preceded by ReLU [27], which aligns RoI.). After that, we use R-CNN bounding boxes with stride-two 2x2 convolution layers, and the MLP of two 1024 large hidden networks

reduces the spatial scale to 7X7. Finally, identical Mask layers are added to each of the K groups to provide a 7X7 mask estimation. The MLP's hidden layers are activated with ReLU, and its outputs are activated with the sigmoid activation function.

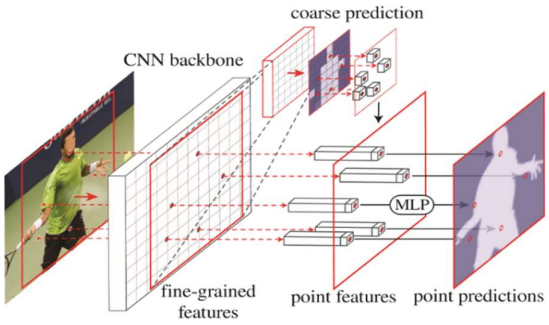


Figure 3: Standard PointRend Segmentation Framework

4.4 PointRend. Using bilinear interpolation N-dimensional function vector is derived by the coarse prediction at each stage. From the FPN P2 step, PointRend captures a 256-dimensional feature vector. In relation to the input image, this level has a stride of 4. It renders a N-class prediction at selected stages using an MLP with three hidden layers and 256 channels. To build the input vector for the next layer, supplement 256 output channels in each layer of the MLP with N coarse prediction feature vectors. It uses Rectified Linear Unit (ReLU) within the Multi-Layer Perceptron (MLP) and sigmoid for output.

4.5 Training. Tails and the regular 1x training plan are also included, and 14^2 points should be analyzed. For PointRend use the skew sample technique and the likelihood of the ground truth class interaction with k=3 and β=0.75. It employs the interval between similar coarse predictions as a measure of pointwise uncertainty. For a predicted box with ground-truth class c, add the binary cross-entropy loss for the c-th MLP contribution over the 142 points. For the mask expected for class c, the compact coarse prediction head uses mean cross-entropy loss, which is similar to the loss used by the baseline 4x conv head. It calculates all loss without re-weighting.

Mask R-CNN combines the bounding boxes and masks head in conjunction at the time of training, but as a cascade during inference. It was discovered that practicing like a cascade which does not boost the baseline of Mask R-CNN, but it can support. PointRend using sampling points within more precise frames, marginally enhancing overall efficiency (~0.2 percent AP, absolute).

4.6 Inference. Unless otherwise stated, inference on a bounding box with each predicted class c, use the adaptive subdivision (at most) of the N=282 most uncertain points based on 224x224 in 5 stages. At each stage, choose and modify techniques to optimize the coarse 7x7 prediction for class c to an average deviation between both the predictions.

4.7 Results.

In Table 1, we compare PointRend and Mask R-CNN Results by its aspect ratios and mean aspect ratios, PointRend outperforms on all datasets. The gap between the COCO categories, Cityscapes and Avenue datasets, which attribute superior sharp point annotation with good quality, irrespective of output resolution, it outperforms to the baseline. The difference between 224×224 and 28×28 is relatively too small because usage of intersection-over-union (IoU) [10] is efficiently biased which concentrate only on object interior pixels values with less sensitive to the its boundary. Visually the difference in boundary pixel quality is shown in Fig. 6.

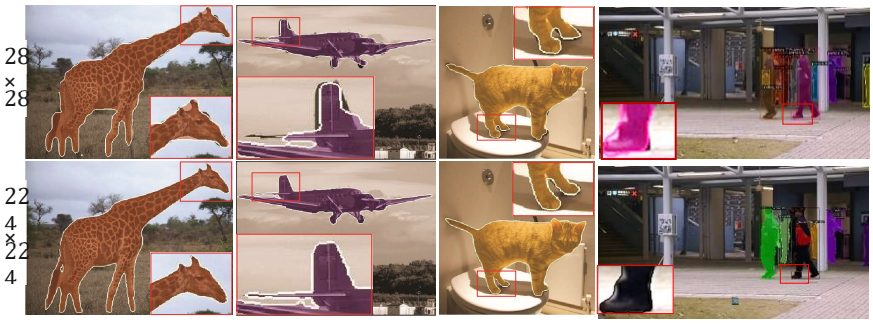


Figure: 4 PointRend Output on different Resolutions

4.7.1 Subdivision inference

Rendering is common challenges faced in pixel labelling and segmentation task because of over and under sampling. This rendering problem solved by using classical iterative subdivision algorithm [37], it adaptively select non-uniform set of points. This subdivision strategy effectively compute high-dimensional segmentation mappings by using an order of magnitude, floating points and dense computations. The major principle or working mechanism of subdivision technique is to follow the location based on huge significantly differ values followed by its neighbor values, reaming locations are taken by interpolating from already computed values is illustrated in figure 7. The subdivision steps followed by points per subdivision in 28X28, 56X56, 112X112, 224X224 resolution with 28^2 points respectively, it has illustrated in figure 6. As in figure 5 saturation of some number of points on different steps of points choose from most inscrutable region first and then remaining areas selected by coarse predictions is sufficient.

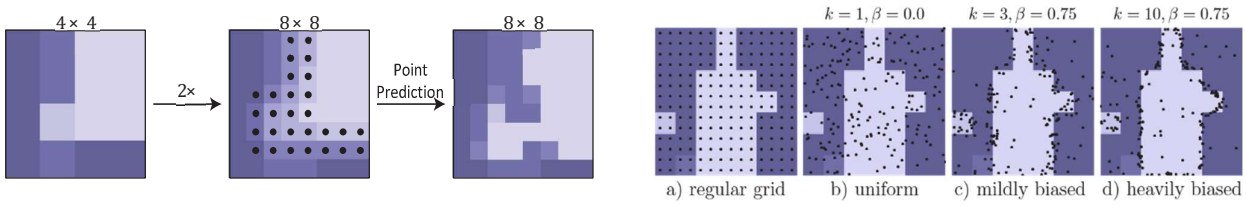
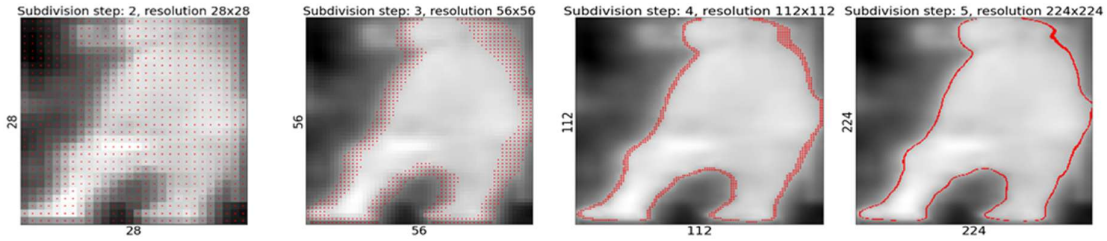


Figure: 5 Saturation of points on various steps.

Figure: 6 Subdivisions steps followed by points.



4.7.2 Instance Predication

The points on each subdivision step are effectively defined using this subdivision and coarse prediction techniques, as well as CNN backbone pre-trained weights. The point features from the input video sequence frames will be returned by this evaluation, and these points will be used to measure the moving object points. These points are subjected to segment-moving instances on each frame. Figure 8 depicts the predicated moving instances. Moving objects are segmented in this figure with the goal of minimizing under sampling in high collision areas. Without oversampling neighbor pixels, each object's boundaries are perfectly segmented.

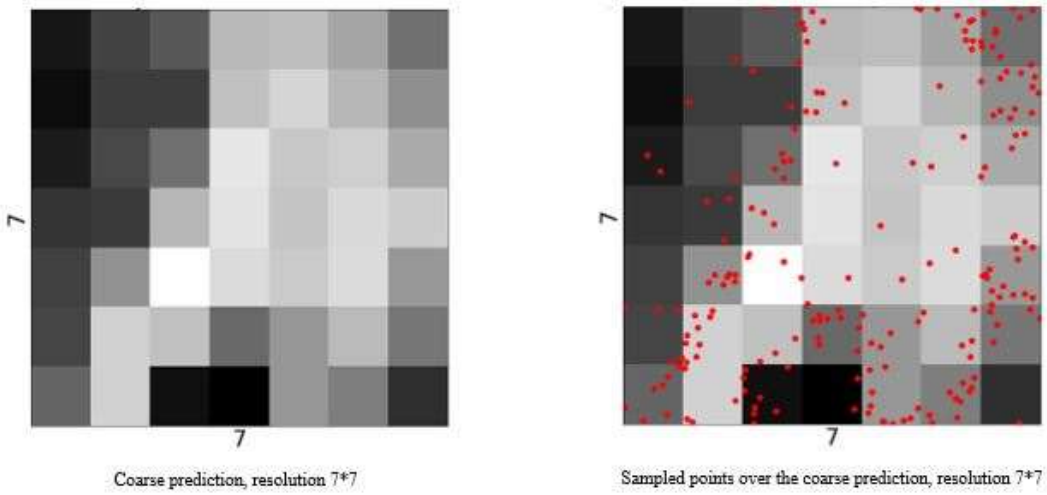


Figure: 7 Coarse prediction resolution with sampled points.

Figure 8: Detection of moving instances on the Avenue dataset.

4.7.3 Key-Frame Extraction

After successful segmentation of instances on every batch on a video sequence of frames using PointRend approach. We need to calculate the minimum instances occluded frames over video sequence. In figure 9, see bottom left instance probability distribution graph (First Graph) shows how instances are saturated among the frame over the each batch. This kind of complex

evaluation with detected moving instances are pooled into EIPM by using algorithm 1, it initialize each frame values. The next frame instances compared by previous selected instance in EIPM by algorithm 2, by this comparison minimum instance inserted into EIPM. In EIPM with its selective frame predication techniques identify the minimum instance, it illustrated in second graph, it shows how minimum instance are selected in each batch. Every batch as listed minimum number of instances. Using these instances values finally least minimum possible instance frames are selected by all batches, these selected frames are called as minimum instance occluded key-frame figure 9 shows selected minimum instance key frames.

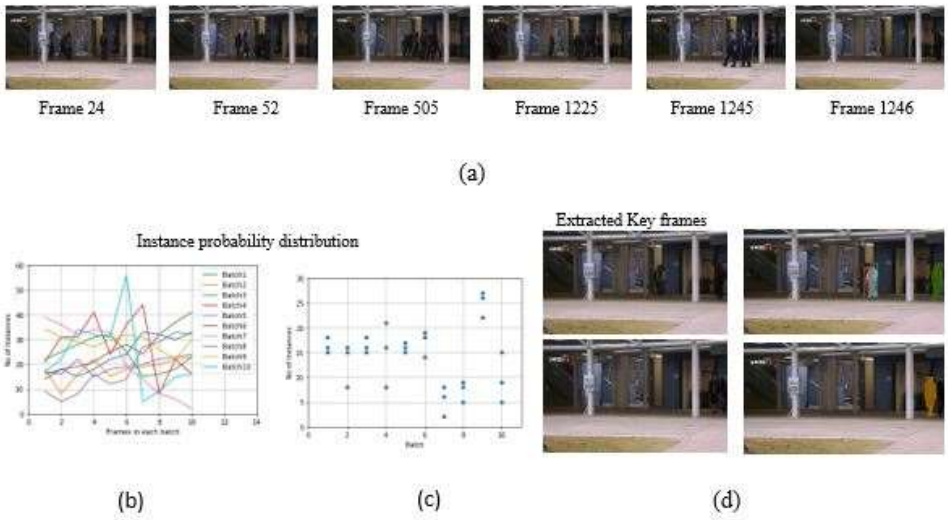


Figure: 9 Least minimum instance key-frame extraction

4.8 Comparison with state-of-the-art methods

Finally, we compare the instance segmentation and key frame extraction on different video dataset. Here we first discuss instance segmentation by taking example of state-of-the-art deep neural architecture Mask-RCNN. Using avenue dataset, we perform the evaluations on instance predications. In Table, 1 shows how instances are segmented on average precision with different IoU' s (intersection over Union) AP@ IoU AP, AP@ IoU AP50, AP@ IoU AP75 and mean average precision (mAP) of both Mask-RCNN and PoinRend segmentation. It clearly shows how each precision vary with respective IoU's over the batch, on each batch mean average precision improves compared to Mask-RCNN. Furthermore, analysis on different benchmark video datasets was performed on UCF-101, HMDB-51, the Table 2 illustrate the overall instance predication accuracy on these datasets, here PointRend gives good results compare to Mask-RCNN Figure 10 depicts the qualitative predicted instances and their aspect resolution, illustrating the differences in segmentation efficiency between Mask-RCNN and PointRend. PointRend retains the resolution as well as the undersampling of densely occluded objects, in this case by precisely masking their boundaries.

BATCH	Mask-RCNN				PointRend			
	AP	AP50	AP75	mAP	AP	AP50	AP75	mAP
1-2	0.31	0.35	0.37	0.343	0.35	0.37	0.33	0.35
2-4	0.34	0.36	0.31	0.336	0.38	0.34	0.31	0.343
4-6	0.34	0.38	0.22	0.313	0.39	0.36	0.34	**0.363**
6-8	0.43	0.22	0.36	0.336	0.37	0.35	0.32	0.346
8-10	0.33	0.31	0.24	0.293	0.38	0.37	0.35	**0.366**

Table 1: Comparison of IoU on two Instance segmentation methods.

Figure: 10 Aspect Resolution of Mask-RCNN vs PointRend Segmentation.

Dataset	PointRend (%)	Mask-RCNN (%)
UCF-101	87	84
HMDB-51	58	51
Avenue	89	83

Table 2: Instance segmentation accuracy comparison on different datasets.

After successful evaluation of instances in these datasets, finding the key frames based on instances. The Table-3 shows variation of instances from batch to batch and differences of

finding key frames instances on PointRend and Mask-RCNN approaches. Here PointRend results good accuracy because of effective instance prediction by using its various subdivision strategy. The error accuracy of each batch computed by differences of least minimum instances predicted in frames. Based on the predications average error accuracy in key frame is significantly decreased by PointRend segmentation.

Average Error in Key Frame		
BATCH	Mask-RCNN	PointRend
1-2	±1.27	±0.838
2-4	±2.45	±0.782
4-6	±2.00	±0.807
6-8	±2.18	±1.500
8-10	±1.88	±0.434
Average Accuracy	**1.956**	**0.9854**

Table 3: Average accuracy of extracted key frame.

5. Conclusion and future work

In this paper, we propose a novel key frame selection approach based on number of instances in a video sequences. Here we majorly concentrate on state-of-the-art instance segmentation approach with lot of attention on highly occluded objects and its boundaries called PointRend instance segmentation. This approach extends to predict instances on image to video. In video overall predicted instances on each frame organized for computation into Effective Instance Predication Matrix (EPIM) with two elect and select strategy for selection and comparison to find least minimum instance occluded key frame. This least minimum key frame extraction leads to many applications on computer vision and deep learning based video analysis. The minimum instance key frames which helps to find more abstract spatial features in densely occulted videos, which improves the background extraction on real-time scenarios also uses Generative Adversarial Neural Networks to solve some inpainting problems. Hence, our future work proceeds to solve these observations.

References

1. Aote SS, Potnurwar A (2019) An automatic video annotation framework based on two level keyframe extraction mechanism. Multimed Tools Appl 78:14465–14484. doi: 10.1007/s11042-018-6826-3
2. Arnab A, Torr PHS (2017) Pixelwise instance segmentation with a dynamically instantiated network. Proc - 30th IEEE Conf Comput Vis Pattern Recognition, CVPR 2017 2017-Janua:879–888. doi: 10.1109/CVPR.2017.100
3. Chaaraoui AA, Climent-Pérez P, Flórez-Revuelta F (2013) Silhouette-based human

action recognition using sequences of key poses. Pattern Recognit Lett 34:1799–1807. doi: 10.1016/j.patrec.2013.01.021

4. Chen DY (2011) Modelling salient visual dynamics in videos. Multimed Tools Appl 53:271–284. doi: 10.1007/s11042-010-0511-5
5. Chen K, Pang J, Wang J, Xiong Y, Li X, Sun S, Feng W, Liu Z, Shi J, Ouyang W, Loy CC, Lin D (2019) Hybrid task cascade for instance segmentation. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 2019-June:4969–4978. doi: 10.1109/CVPR.2019.00511
6. Chen X, Girshick R, He K, Dollar P (2019) TensorMask: A foundation for dense object segmentation. Proc IEEE Int Conf Comput Vis 2019-October:2061–2069. doi: 10.1109/ICCV.2019.00215
7. Colleges GTUA, Academy O, Academy O, Academy O, Science AC, Technology I, Science AC (2014) Microsoft COCO. Eccv 740–755
8. Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B (2016) The Cityscapes Dataset for Semantic Urban Scene Understanding. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 2016-December:3213–3223. doi: 10.1109/CVPR.2016.350
9. Dou J, Qin Q, Tu Z (2019) Background subtraction based on deep convolutional neural networks features. Multimed Tools Appl 78:14549–14571. doi: 10.1007/s11042-018-6854-z
10. Everingham M, Eslami SMA, Van Gool L, Williams CKI, Winn J, Zisserman A (2015) The Pascal Visual Object Classes Challenge: A Retrospective. Int J Comput Vis 111:98–136. doi: 10.1007/s11263-014-0733-5
11. Fooladgar F, Kasaei S (2020) A survey on indoor RGB-D semantic segmentation: from hand-crafted features to deep convolutional neural networks. Multimed Tools Appl 79:4499–4524. doi: 10.1007/s11042-019-7684-3
12. De Geus D, Meletis P, Dubbelman G (2019) Single network panoptic segmentation for street scene understanding. IEEE Intell Veh Symp Proc 2019-June:709–715. doi: 10.1109/IVS.2019.8813788
13. Gutoski M, Aquino NMR, Ribeiro M, Lazzaretti AE, Lopes HS (2019) Detection of Video Anomalies Using Convolutional Autoencoders and One-Class Support Vector Machines. 1–12. doi: 10.21528/cbic2017-49
14. He K, Gkioxari G, Dollar P, Girshick R (2017) Mask R-CNN. Proc IEEE Int Conf Comput Vis 2017-Octob:2980–2988. doi: 10.1109/ICCV.2017.322
15. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp 770–778
16. Ilg E, Mayer N, Saikia T, Keuper M, Dosovitskiy A, Brox T (2017) FlowNet 2.0: Evolution of optical flow estimation with deep networks. Proc - 30th IEEE Conf Comput Vis Pattern Recognition, CVPR 2017 2017-Janua:1647–1655. doi: 10.1109/CVPR.2017.179
17. Jeong J, Shin J (2019) Training CNNs with Selective Allocation of Channels
18. Kirillov A, Levinkov E, Andres B, Savchynskyy B, Rother C (2017) InstanceCut: From

edges to instances with MultiCut. Proc - 30th IEEE Conf Comput Vis Pattern Recognition, CVPR 2017 2017-Janua:7322–7331. doi: 10.1109/CVPR.2017.774
19. Kirillov A, Wu Y, He K, Girshick R (2019) PointRend: Image segmentation as rendering. arXiv 1–11
20. Kulhare S, Sah S, Pillai S, Ptucha R (2016) Key frame extraction for salient activity recognition. Proc - Int Conf Pattern Recognit 0:835–840. doi: 10.1109/ICPR.2016.7899739
21. Laptev I, Lindeberg T (2003) Space-time Interest Points Ivan. In: Ninth IEEE International Conference on Computer Vision (ICCV'03) 0-7695-1950-4/03. pp 0–7
22. Lecun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521:436–444
23. Liu S, Jia J, Fidler S, Urtasun R (2017) SGN: Sequential Grouping Networks for Instance Segmentation. Proc IEEE Int Conf Comput Vis 2017-Octob:3516–3524. doi: 10.1109/ICCV.2017.378
24. Liu S, Qi L, Qin H, Shi J, Jia J (2018) Path Aggregation Network for Instance Segmentation. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 8759–8768. doi: 10.1109/CVPR.2018.00913
25. Mademlis I, Tefas A, Pitas I (2018) A salient dictionary learning framework for activity video summarization via key-frame extraction. Inf Sci (Ny) 432:319–331. doi: 10.1016/j.ins.2017.12.020
26. Mescheder L, Oechsle M, Niemeyer M, Nowozin S, Geiger A (2019) Occupancy networks: Learning 3D reconstruction in function space. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 2019-June:4455–4465. doi: 10.1109/CVPR.2019.00459
27. Nair V, Hinton GE (2010) Rectified Linear Units Improve Restricted Boltzmann Machines. In: Fürnkranz J, Joachims T (eds) Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel. Omnipress, pp 807–814
28. Palmer R, West G, Tan T (2012) Scale proportionate histograms of oriented gradients for object detection in co-registered visual and range data. 2012 Int Conf Digit Image Comput Tech Appl DICTA 2012. doi: 10.1109/DICTA.2012.6411699
29. Ray KS, Asari VK, Chakraborty S (2017) Object Detection by Spatio-Temporal Analysis and Tracking of the Detected Objects in a Video with Variable Background. 1–13
30. Shelhamer E, Long J, Darrell T (2017) Fully Convolutional Networks for Semantic Segmentation. IEEE Trans Pattern Anal Mach Intell 39:640–651. doi: 10.1109/TPAMI.2016.2572683
31. Singh RD, Mittal A, Bhatia RK (2019) 3D convolutional neural network for object recognition: a review. Multimed Tools Appl 78:15951–15995. doi: 10.1007/s11042-018-6912-6
32. Sun J, Wang J, Yeh T-C (2017) Video Understanding: From Video Classification to Captioning. 1–9
33. Suresha M, Kuppa S, Raghukumar DS (2020) A study on deep learning spatiotemporal models and feature extraction techniques for video understanding. Int J Multimed Inf Retr. doi: 10.1007/s13735-019-00190-x

34. Tang H, Liu H, Xiao W, Sebe N (2019) Fast and robust dynamic hand gesture recognition via key frames extraction and feature fusion. Neurocomputing 331:424–433. doi: 10.1016/j.neucom.2018.11.038
35. Wang H, Kl A, Schmid C, Cheng-lin L, Wang H, Kl A, Schmid C, Recognition LCA, Kl A (2011) Action Recognition by Dense Trajectories To cite this version : Cvpr'11
36. Wang T, Delaunay-lms IC (2012) Histograms of Optical Flow Orientation for Visual Abnormal Events Detection. IEEE Ninth Int Conf Adv Video Signal-Based Surveill 13–18. doi: 10.1109/AVSS.2012.39
37. Whitted T (1979) An improved illumination model for shaded display. ACM SIGGRAPH Comput Graph 13:14. doi: 10.1145/965103.807419
38. Wu Z, Wang X, Jiang Y, Ye H, Xue X (2015) Modeling Spatial-Temporal Clues in a Hybrid Deep Learning Framework for Video Classification. In: MM'15 Proceedings of the 23rd ACM international conference on Multimedia. pp 461–470
39. Wu Z, Yao T, Fu Y, Jiang Y-G (2017) Deep learning for video classification and captioning. In: Frontiers of Multimedia Research. pp 3–29
40. Zhang H, Goodfellow I, Metaxas D, Odena A (2018) Self-Attention Generative Adversarial Networks