Санкт-Петербургский государственный университет

Программная инженерия

Немчинов Егор Игоревич

# Сегментация изображений человека в видеопотоке со статичным фоном

Бакалаврская работа

Научный руководитель:
ст. преп., к.ф.-м.н Д. В. Луцив

Рецензент:
Специалист по машинному обучению
AI Factory, Inc.
Ю. В. Волков

Санкт-Петербург
2020

SAINT PETERSBURG STATE UNIVERSITY

Software engineering

Egor Nemchinov

# Human instance segmentation from video with a static background

Bachelor's Thesis

Scientific supervisor:
Sr. lecturer, PhD Dmitry Luciv

Reviewer:
Machine learning engineer
AI Factory, Inc.
Yurii Volkov

Saint Petersburg
2020

# Contents

# Introduction

Background removal from a video of people with a static background is a common problem for different tasks such as background replacement, 3D motion capture, video surveillance and video analysis of human activities [5].

Such formulation of background removal task may also be actively used for cheaper segmentation of videos with people, since with the rapid growth of Artificial Intelligence models nowadays, the amount of data needed also increases incredibly fast [49]. Labelling of datasets usually requires a lot of effort, which translates into money and time for companies striving to train neural networks for previously unsolved problems. The cost of creating a new dataset in Computer Vision largely depends on a type of problem. For instance, it is easier and thus, cheaper to assign one of a few classes to the image than to draw a semantic segmentation map.

Quality and speed of human segmentation from a single image has been improved in a significant manner in the last few years. However, when directly adopting a deep human segmentation model to the task of video human segmentation, the performance suffers from a few problems such as discontinuity of video frames and the speed of segmentation process [46]. Additionally, applying human segmentation from a single image discards a lot of very important temporal information, since every next frame is very similar to the previous ones and is semantically connected to them. Thus, an approach of video human segmentation must be developed with these issues in mind.

This work considers a special case of videos with a static background, which is explored by a field of background subtraction. However, existing methods of background subtraction mostly aren't specialized to a certain class of objects, while human segmentation approaches aren't made for a specific case of a static background. Of course, taking a picture of the background before filming a video with a person can eliminate the need for the first part, but in reality, it's not always possible, thus the problem of calculating the background should be solved.

Additionally, there is a problem of high barrier to entry for segmentation approaches. The environment for open-source approach has to be set up, which may be a hard problem not only for, for instance, video editors, but even for programmers specialized in this area.

Considering all the observations, the task of human segmentation from videos with a static background may give more accurate results than both human segmentation approaches and general background subtraction methods. Thus, the scope of this work is the development of technology to make human segmentation from a video with a static background and implementation of the server-based solution with web-interface to process videos to simplify usage for a common user.

# 1 Problem statement

The goal of this work is to build a system for segmentation of human videos with a static background, which would be available for a common user.

This work can be divided into following tasks:

1. conduct a survey of the field;
2. develop a method for segmentation of human videos with a static background;
3. make a quantitative comparison with other approaches and perform an analysis;
4. create web-service with implementation of the method.

# 2 Survey of the field

This work is closely related to two topics in computer vision: *Human segmentation* and *Background subtraction.* This section gives an overview of these fields and existing approaches.

## 2.1 Human segmentation

Topics of human segmentation have been explored deeper due to the presence of high-demand real-life applications such as video surveillance, action localization, pedestrian detection, virtual-reality simulation and 3D human modelling [34]. The goal of human segmentation is to identify a human in an image or video and separate it from the background.

Firstly, human pose estimation methods are overviewed since this class-specific information is used to get key information about human objects and may be used as one of the inputs for segmentation methods. Then, human segmentation methods are surveyed. And, finally, human matting methods allow for a very exact ”soft” segmentation to separate a person from the background accurately. Finally, available datasets for these tasks relevant for our problem are surveyed as they may be useful for the process of developing a new approach.

### 2.1.1 Human pose estimation

An overview of 2D human pose estimation methods is given in [8]. 2D human pose estimation methods can be divided into two categories: bottom-up and top-down methods. Top-down methods firstly detect people and then figure out their keypoints, while bottom-up keypoints start with low-level pixel evidence, detect separate body parts of all people in the image and then group them together. Methods are evaluated and compared on COCO dataset [29], where more than 200k human images are labeled with a skeleton consisting of 17 joints with visibility.

Out of top-down methods, [10] is a leading method with an open implementation [1], while AlphaPose implementation[2] performs better on PoseTrack 2017 dataset by MOTA (Multiple Object Tracking Accuracy) metric. Bottom-up methods include leading on COCO method MultiPoseNet [22], although it lacks an open-source implementation, while OpenPose [31] has a real-time open-source implementation[3].

### 2.1.2 Segmentation

A wide variety of human segmentation approaches exist, an overview of the field is given in the literature review of methods for human segmentation on static background [45].

There are approaches that address wider problem scopes like semantic segmentation, i.e. multi-class classification for each pixel of the image. Since binary classification problem scope is a subset of multiclass classification set of problems, human segmentation could be performed utilizing only human class with semantic segmentation. Currently, state-of-the-art approach for semantic segmentation on both PASCAL VOC 2012 test and Cityscapes [48] datasets is DeepLabv3+ [13].

Mask-RCNN [28] is a method that simultaneously detects and performs segmentation of people. It operates on any input resolution, but segmentation results of Mask-RCNN largely depend on the quality of its detection.

In pose-based human segmentation approaches like [37], [38] and [33], neural network conditions on given person pose keypoints to infer a segmentation. In [33] detection of human body key parts is done as a first step and segmentation is calculated based on locations of these key parts. Pose estimation can be done with OpenPose [31], as suggested in [45], or, alternatively, AlphaPose [35]. *Pose2Seg* shows state-of-the-art results for human instance segmentation problem on COCOPerson dataset (part containing people from COCO dataset) [29], outperforming detection-based methods like Mask-RCNN. In Fig. (1) structure of *Pose2Seg* pipeline is shown. Firstly, feature extraction is performed, and after that the pipeline

[1] https://github.com/leoxiaobin/deep-high-resolution-net.pytorch

[2] https://github.com/MVIG-SJTU/AlphaPose

[3] https://github.com/CMU-Perceptual-Computing-Lab/openpose

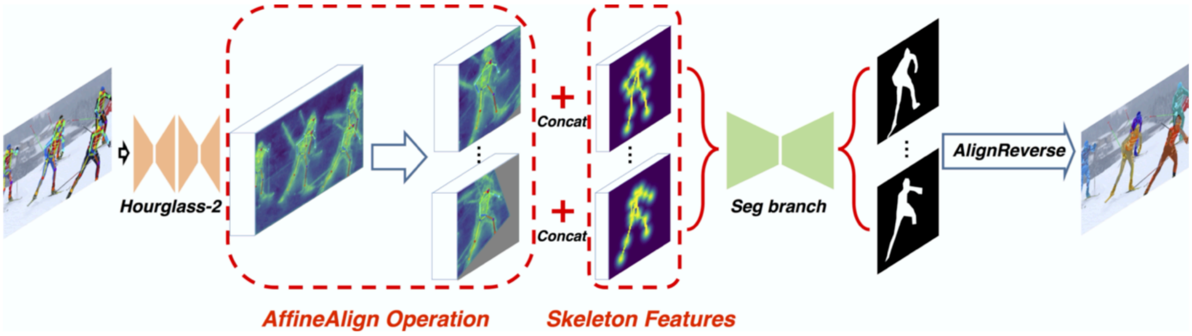

Figure 1: Flowchart of Pose2Seg [46]

is ran independently for each selected person. AffineAlign operation makes alignment of each person to match closely template human poses, then skeleton features are drawn and concatenated with feature maps extracted by a base layer. Skeleton features include part affinity fields, representing human pose skeleton structure and part confidence maps highlighting main joints. Part affinity fields, first introduced in [31], contain information about location and orientation of human limbs. After these steps, SegModule is applied to get final segmentation masks, which are then transformed back into the original image.

### 2.1.3 Matting

Additionally, there are methods of matting – estimating per-pixel foreground color and alpha. Many methods may require trimaps as an input, i.e. images where pixels are marked as one of three classes: background, foreground and uncertainty. It would be best for trimaps to be created manually, but as an option, explored by [3], trimaps can be estimated by thresholding probability of a certain class at inference of segmentation neural networks, which worked for them better than another option of erosion, dilation and then blurring of approximate binary mask received from segmentation methods.

Over years, methods of matting, more specific to the problem of this work methods have been developed: video matting, matting with known natural background and human matting. Some traditional approaches like

Poisson matting [32] and Bayesian matting [47] may handle known background but also require trimaps. Video-specific matting methods may utilize optical flow [41] – pattern of apparent motion of image objects between two consecutive frames – and, optionally, known background [44]. And, finally, there are a few human matting approaches: [15] performs portrait matting using segmentation cues, while [46] performs trimap-free matting for whole bodies.

State-of-the-art human matting approach [3] combines all of these possible specifics: it's a trimap-free human matting algorithm, that utilizes known background. This work is closely related to the problem of this work and has some relevant ideas, thus it will be studied a bit closer.

Background matting requires a few images as the input, which are: source image with a person over a background, background image, rough person mask and a few temporally adjacent frames in case of video segmentation. In that work, training is split into two parts: firstly network is trained supervisely on the Adobe Matting Dataset [11], then copy of this network is trained unsupervisely using original network as a teacher and utilizing learned discriminator. Unsupervised training allows to overcome domain gap between objects of Adobe Matting Dataset and target domain – human bodies in this case. It is recommended that background is captured separately with the same focal length, exposure and that background does not have any dynamic elements, i.e. fully a static picture. The output of the network is matte and foreground image: it is argued, that only alpha may not be enough for accurate background removal. Flowchart of the method is shown in Fig (2).

### 2.1.4 Datasets

Microsoft Common Objects in Context (COCO) [29] is a large-scale dataset that contains more than 200,000 images with 250,000 labeled person instances. Usage of COCO covers image captioning and keypoints detection. COCO Keypoint Detection Challenge of 2016 and 2017 aims to capture keypoints of people, for achieving such goal it includes annotations for each person which include 17 body joints and instance human body seg-

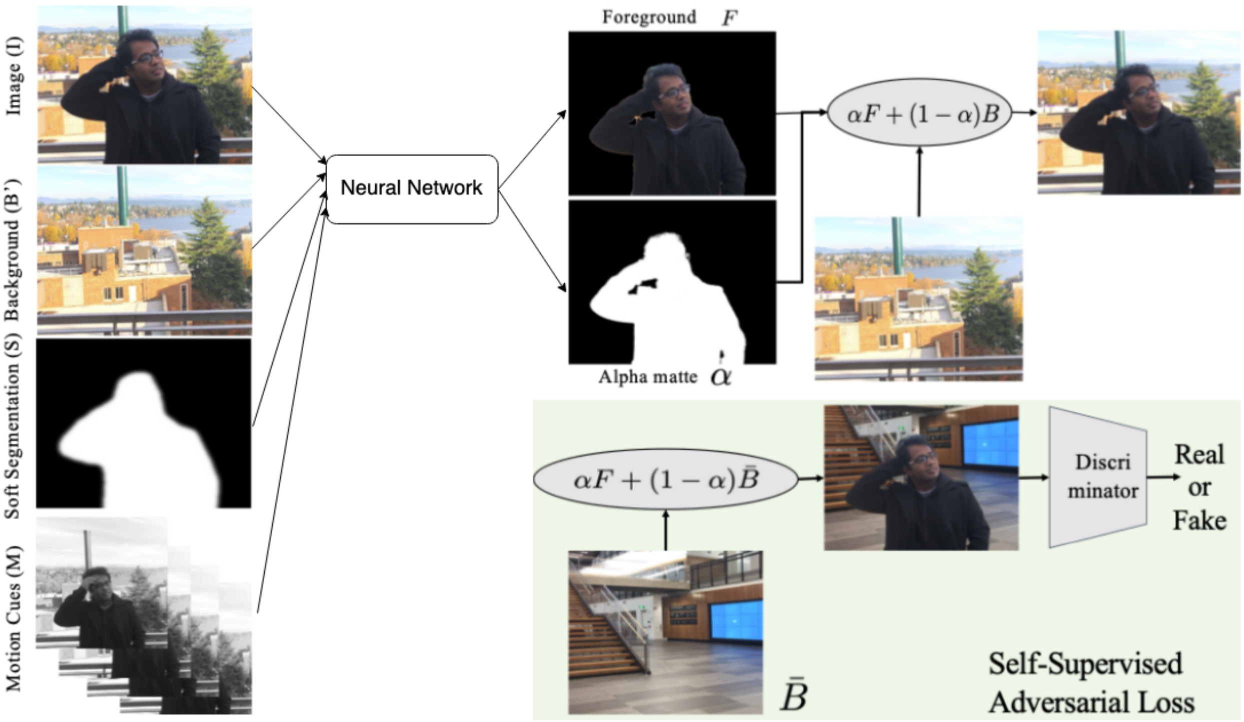

Figure 2: Flowchart of Background-Matting, parts of the flowchart are taken from [3]. Input images on the left, on the top right outputs of the network and composition onto the given background to calculate loss, on the bottom images are merged onto another background for loss from discriminator network.

mentation. OCHuman dataset presented in [33]. The dataset contains 4731 images with 8110 human instances. It's focused on complex cases of human occlusion and recommended by the authors to be used for validation and testing to check robustness of an approach.

## 2.2 Background subtraction

Problem solved in this work may be classified as a problem of Background subtraction, which is considered to be a subset of Video object segmentation and tracking (VOST) methods.

First, an overview of VOST is given to see whether other fields may be useful to the task of this work. Then methods of background subtraction and background estimation are overviewed.

### 2.2.1 Video object segmentation and tracking

Object segmentation and object tracking are a fundamental research area in the computer vision community. Both have difficulties with handling occlusion, deformation, motion blur and scale variation, but each one has its specifics. In [43] authors give an overview of currently existing methods of video object segmentation and tracking (VOST), the overall scheme is shown in Fig (3). The top two branches, i.e. Unsupervised VOS methods and Semi-supervised VOS methods are relevant for the problem of this work. In [12] a systematic review of methods for background subtraction based on deep neural networks is given.

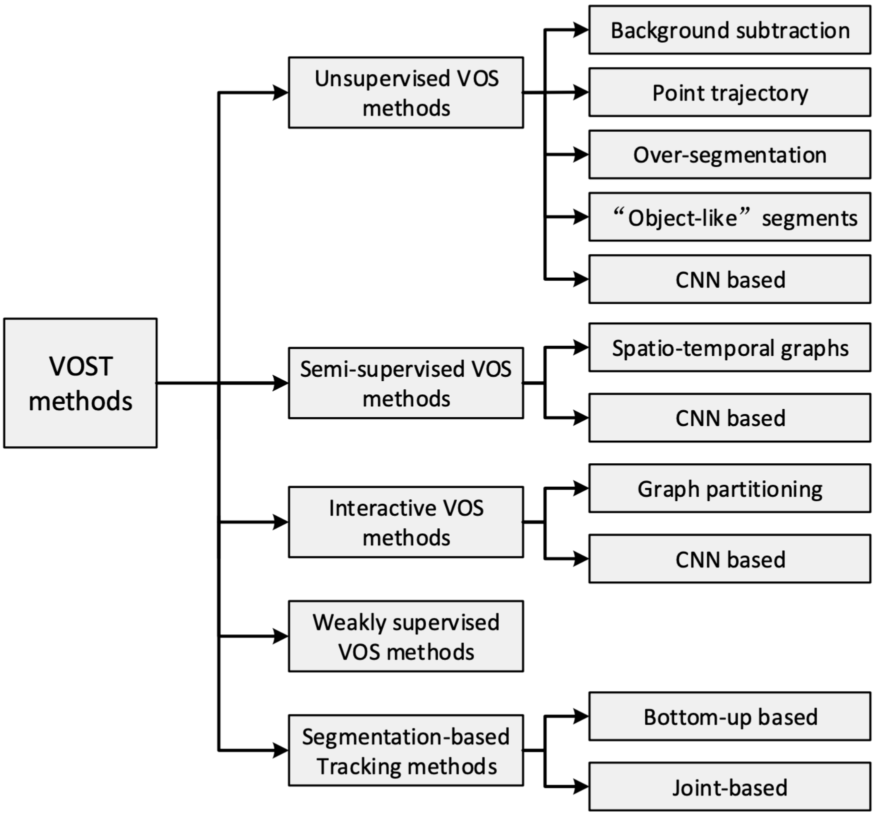

Figure 3: Taxonomy of video object segmentation and tracking from [43]

### 2.2.2 Background subtraction methods

Background subtraction methods consider rapidly changing pixels as foreground and simulates the background appearance of each pixel. Though background subtraction methods can simulate backgrounds in 2D or 3D motions, the focus here is on stationary background models [4, 18, 40].

An overview of background subtraction methods for real applications is given in [5]. Performance of different background subtraction methods, which are not based on deep learning, may largely vary depending on the use case and specifics of conditions. That being said, background subtraction algorithms with available implementations described in [5] include algorithms based on MOG (Mixture of Gaussians) from OpenCV [4] and BGSLibrary [5] – C++ framework that includes more than 43 background subtraction algorithms.

Background subtraction methods based on deep learning are surveyed in [12], where they are evaluated on CDNet 2014 [6] dataset and compared. Out of non-parametric methods the leading one is SuBSENSE [39], which doesn't require adaptation to every specific case and handles well illumination changes and shadows. Methods that have to be fine-tuned on a single mask or a few masks, like [27] give the best results on CDNet 2014, but won't be considered as the data for fine-tuning on a single mask is rarely available and is computationally expensive. BScGAN [1] is the best unsupervised method based on GAN: generator takes an image with foreground and background image, then learns to generate a mask, while discriminator learns to distinguish ground truth mask from generated ones.

Another class of methods is developed to train a network to incorporate optical flow [20, 26, 30, 36], which is usually generated with FlowNet method [17]. Some methods employ a Recurrent Neural Network (RNN) for modelling mask propagation with optical flow [19, 26]. MaskTrack [25] method trains a refine the previous frame mask to create the current frame mask, and directly infer the results from optical flow, pipeline of which one can find in Fig. (4). An approach based on Spatio-Temporal GANs [16] shows very good accuracy with quick inference, which is achieved by using a big temporal window with two discriminators during training and only using a generator with a temporal window of size 2 during inference.

[4] https://opencv.org

[5] https://github.com/andrewssobral/bgslibrary

### 2.2.3 Background estimation

In cases where only background image itself is needed, methods of background estimation may be useful. Experimental results for background estimation on SBMnet [14] dataset task are given in [12]. The best result on that dataset is achieved by [2], followed by algorithms with available online implementations: LabGen-p[6] [23] and LabGen-of[7] [24] algorithm .

### 2.2.4 Datasets

As follows from [21], the most popular dataset for background subtraction task by amount of citations is CDnet 2012 [7]. The CDnet 2012 [7] dataset was recorded in 2012 with distinct cameras including PTZ camera, low-resolution IP cameras, mil-resolution camcorders, and thermal cameras. It consists of 31 videos having total 90,000 video frames and is grouped into six categories to cover a wide range of challenges that exist in most video analytics applications. Later, CDnet 2014 [6] was released with 22 additional videos. It contains approximately 10 videos containing only people and results of other approaches can be seen for each video, thus a subset of this dataset can be used in order to .

Additionally, The Labeled and Annotated Sequences for Integral Evaluation of Segmentation Algorithms (LASIESTA) [9] is a dataset for background subtraction with most of the videos containing only human objects. Recorded in 2016, it contains a collection of 48 videos recorded with mostly static cameras in indoor and outdoor scenarios. The challenges in the dataset include shadows, dynamic background, illumination changes, occlusion, camouflage, moving camera, bootstrapping, stationary fore- ground objects, and challenging weather.

## 2.3 Survey conclusions

The problem of human segmentation from videos with a static background, while being on the intersection of a few fields, doesn't fully match

[6] https://github.com/benlaug/labgen-p

[7] https://github.com/benlaug/labgen-of

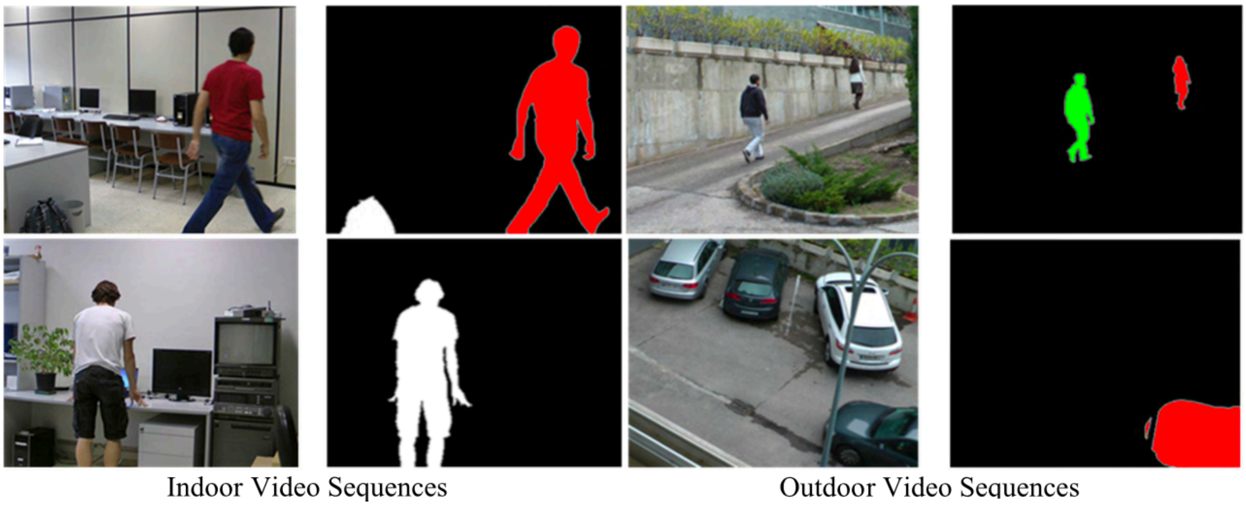

Figure 4: Examples from scenes in LASIESTA dataset [9]

any specific solution: either a background image must be provided separately [3], or information about human class is underutilized as it happens to be in most generic approaches for background subtraction [12], or human body segmentation doesn't take into account information about known background or neighbor frames.

Background Matting [3] pipeline may be used as a foundation and improved further. For human segmentation, Pose2Seg [46] may be taken as it is very stable even in sophisticated conditions; it gives a good rough approximation, but in such segmentations lack of details is present. Then, information about human pose may be added as an additional input to Background-Matting network architecture, which wouldn't cost additional computations as it already has been calculated for Pose2Seg masks inference. And finally, instead of capturing static background during the video filming stage, it may be calculated using one of background calculation methods: either LabGen [23, 24], or a class-specific background calculation method.

# 3 Implementation

Method developed in this work may be considered class-specific background subtraction method since it focuses on taking into account specifics of human class for background subtraction task.

This chapter describes pipeline for the developed solution in general and then it's modules in details.

## 3.1 Method overview

Pipeline consists of, firstly, creating segmentation masks and pose keypoints for all frames, then estimation of static background, and, lastly, all these source frames, keypoints, segmentation masks and static background are given as an input to a modified version of Background Matting network. Described pipeline can be seen on Fig. (5). Whole pipeline is implemented in Python language.

Reusing human pose keypoints allows for better utilization of information specific to human class, but makes the pipeline sensitive to the quality of keypoint detection, therefore choice of keypoints detector has a big impact on the quality of output. Segmentation masks are also reused for both background estimation step and the final step of mask refinement, thus utilizing human class-specific information in as many steps as possible.

## 3.2 Human segmentation

A few approaches have been tested for human segmentation task. Utilizing human class segmentation from DeepLabv3+ was used as a baseline. Additionally, Pose2Seg approach was tested on top of both OpenPose keypoints and AlphaPose keypoints: these were selected as popular open-source implementations of both bottom-up and top-down methods of 2D human pose estimation, which are nearly state-of-the-art in terms of quality and perform well in terms of speed. Pose2Seg has state of the art human segmentation results on COCO dataset and has open-source implementation

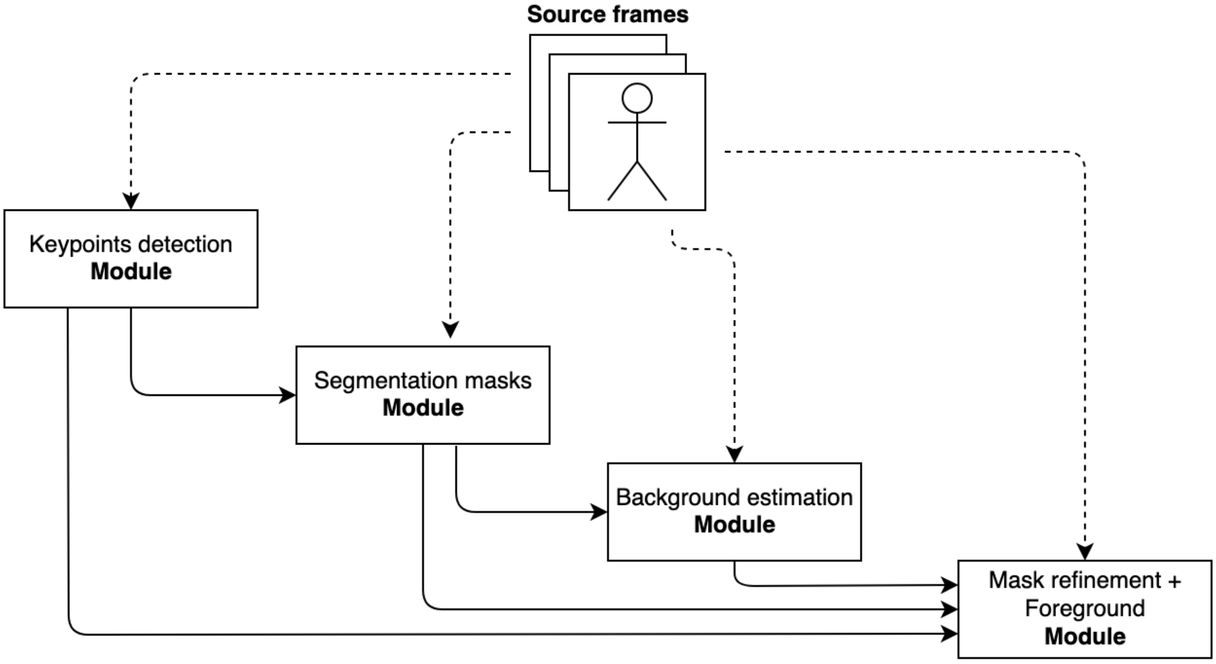

Figure 5: Flowchart of developed method's pipeline

[8].

Comparison was made on subset of LASIESTA containing videos with only people using F-measure as a metric. To infer DeepLabv3+ masks a script used by [3] was used, which runs DeepLabv3+ model and thresholds probability of human class at value 0.95, thus getting binary masks for human class. For Pose2Seg, however, the process consists of two steps. Firstly, keypoints from AlphaPose from open-source implementation [9] and OpenPose from open-source implementation [10] were predicted for all videos of the subset. Then Pose2Seg was supposed to be ran utilizing these keypoints as a part of the input. A script had to be written in order for Pose2Seg to infer segmentation masks using custom keypoints.

As one can see in the table (1), Pose2Seg gives the best results of all in recall and f-measure when used with AlphaPose keypoints. Dependence on keypoints seems to influence only recall metric, precision stays the same. That's possibly because what really matters for segmentation is not accu-

[8] https://github.com/liruilong940607/Pose2Seg

[9] https://github.com/MVIG-SJTU/AlphaPose

[10] https://github.com/CMU-Perceptual-Computing-Lab/openpose

racy of keypoints but whether all people were found on these frames. OpenPose detector seems to miss people on the frames more often than AlphaPose, therefore recall metric is lower. DeepLabv3+ shows better precision than Pose2Seg, especially in occluded scenarios (”I_OC_01”, ”I_OC_02”). For the solution described in this work, recall is more important since there is a step of mask refinement, in which irrelevant parts of mask can be filtered. Thus, the combination of Pose2Seg with AlphaPose were chosen as their recall and F-measure are generally higher.

Though in rare specific cases, e.g. on a black background it works with good accuracy, in most cases it draws rough segmentation masks. Although these masks serve as a good first approximation, they need to be largely refined.

| Method | Mean precision | Mean recall | Mean F-measure |
|---|---|---|---|
| Deeplabv3+ | **0.96** | 0.95 | 0.95 |
| Pose2Seg (OpenPose) | 0.94 | 0.91 | 0.92 |
| Pose2Seg (AlphaPose) | 0.94 | **0.98** | **0.96** |

Table 1: Comparison of segmentation results on a subset of LASIESTA dataset.

## 3.3 Background estimation

For background calculation task, two approaches have been tested against each other on a subset of videos from SBMnet dataset. Only 19 videos that contain a person were taken, categories with camera motion or jitter and illumination changes were excluded from the comparison since our method is not supposed to work with those cases.

First approach includes algorithms by LabGen’s group [23,24] with open-source implementations[11,12].

Another class-specific approach for calculating background is developed in this work, which is calculated using masks for all videos calculated by

[11] https://github.com/benlaug/labgen-p

[12] https://github.com/benlaug/labgen-of

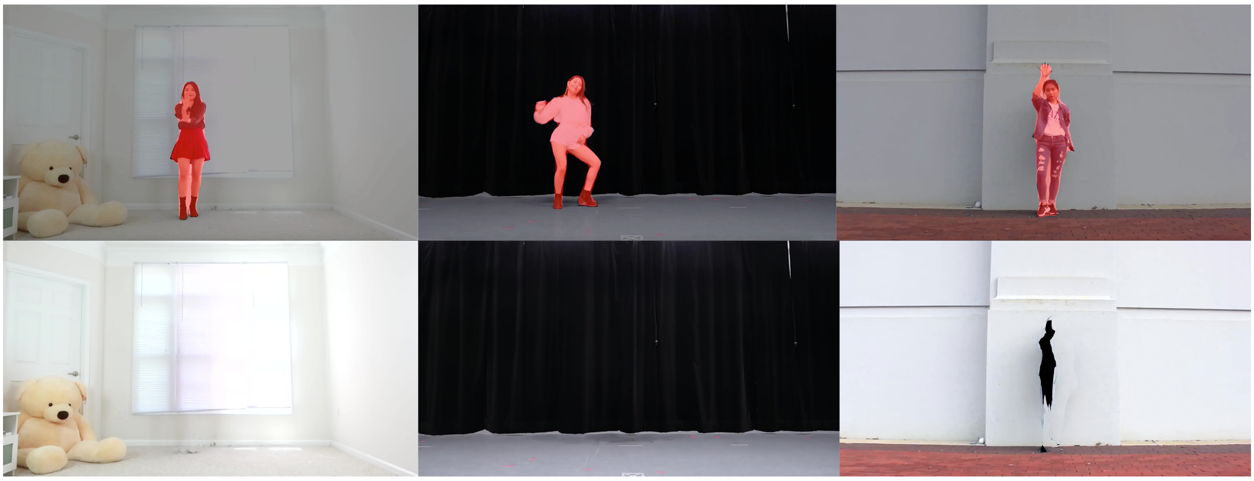

Figure 6: Top row: Pose2Seg [46] segmentations, bottom row: backgrounds calculated from the videos using Pose2Seg segmentation. Middle column shows exceptionally good results of Pose2Seg in case of black background. Right column contains an example of background with unseen part in the middle

segmentation algorithm. It's based on the observation that Pose2Seg masks have high recall, which is discussed in a previous section. For each pair of coordinates on the frame, the median colour was calculated among the frames where this pixel is considered background. This way for some part of the image, where segmentation algorithm has never predicted a human mask, the median or average is taken among all the frames. At the same time it's also possible for some pixels to have no available frames where they would be considered background – in such cases an inpainting algorithm can be used. Having explicit information about lacking parts of the background may be useful – for example, we may know that this part must always be foreground. This algorithm offers a simple and fast solution that utilizes information about an object in the form of segmentation masks. Results of such background calculation are shown in Fig. (6), bottom row.

Through visual comparison on many examples, no big difference was found between these approaches, but the method based on median or average works faster and has a parameter for amount of frames to be sampled, thus it's performance-quality balance can be tuned.

Though no quantitative comparison was performed, it is suggested, that background extraction method doesn't have a big impact on the result of the pipeline. However, it may be considered as a room for improvement.

## 3.4 Mask refinement

As the final step of the pipeline, a modified version of Background Matting was taken to predict an accurate matte of people based on the few inputs computed on the previous steps. Changes to the network, training process and data used for training, are described further.

### 3.4.1 Network changes

Network architecture of original Background Matting can be described briefly. For all inputs there is a block *Prior Encoder* that encodes it into a feature map, then feature maps from each *Prior Encoder* are concatenated separately with feature maps from Image Encoder and pass into a *Selector* block. Then, features from all *Selector* blocks go into *Combinator* block, where final feature tensor is calculated. Finally, network with residual blocks infers alpha matte and foreground image. Adversarial loss is learned to distinguish real images from the images that are segmented and composed onto other backgrounds.

The only change to the architecture of the network is an additional input inspired by Pose2Seg [46] network architecture. Human pose keypoints are loaded and transformed from coordinates of 17 joints for each person into a 55-channel map. These 55 channels consist of Part Affinity Fields (PAFs) taking up 38 channels – 2 channel for each limb – which represent orientation and location of body parts and 17 channels representing heatmaps for each joint are stored.

Modified architecture can be seen in Fig (7). Keypoint maps are processed the same way the other inputs are processed. Firstly, they are encoded using Prior Encoder for keypoints, then these features are stacked with image features and these stacked features are given to a separate Selector block, output of which finally goes into the part of the network that remained the same. To summarize, a new input with keypoint maps is processed similarly to other inputs, then stacked with features from Selector blocks of other inputs.

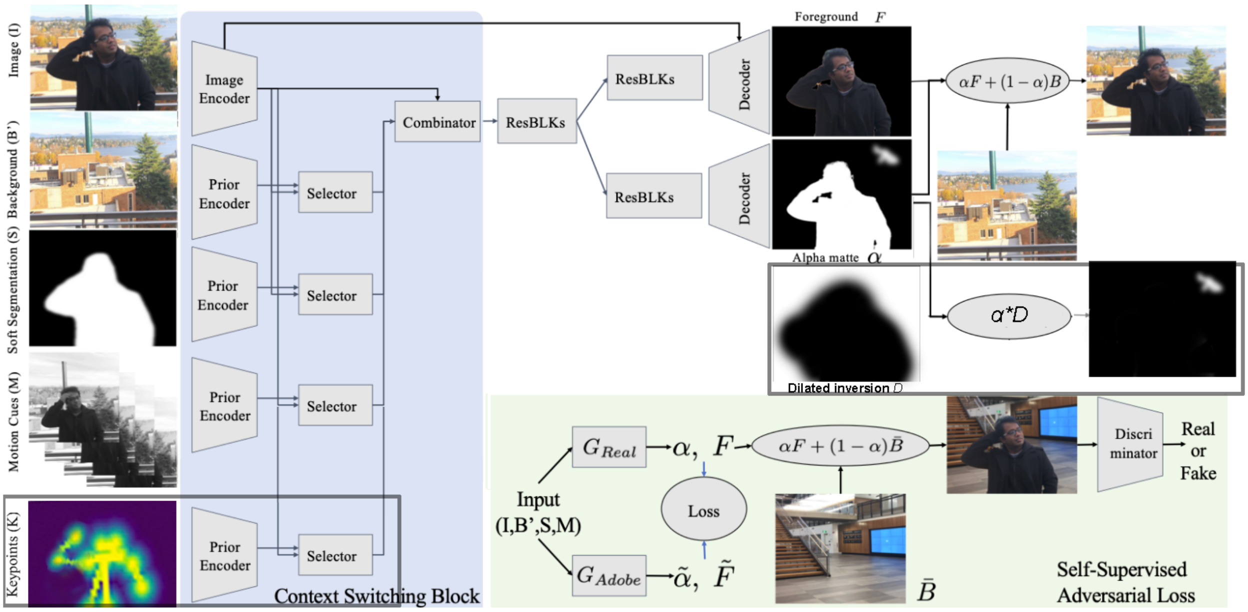

Figure 7: Network architecture of the modified version of Background Matting. Modified parts are outlines with bold gray lines. In bottom left corner is added input with keypoints, in middle right is a divergence loss.

### 3.4.2 Data preparation

In order to teach the network unsupervisely to make accurate segmentation of full human bodies on a static background, a dataset of videos had to be collected. Since the training for bridging domain gap in Background Matting network is performed unsupervisely, data did not require ground truth masks. 40 videos with dancing people on static background were collected from Youtube[13] using open-source library with Command Line Interface youtube-dl[14].

These videos were then trimmed to leave out introductory and final parts as they often included unwanted elements like changing background, popping-up subtitles etc. Then videos were split into frames using FFmpeg[15] [42], processed with AlphaPose and Pose2Seg after which, lastly, back-

[13] https://youtube.com

[14] https://github.com/ytdl-org/youtube-dl

[15] https://ffmpeg.org

grounds were extracted. For testing stage 10% of videos, i.e. 4 out of 40, were left to visually compare training results by quality. Example images from collected videos with predicted masks and calculated backgrounds are shown in Fig. ( 6).

Additionally videos used for training in Background Matting [3] were taken, of which there are a total of 21 videos, plus additional 18 videos for testing. Together, there was a total of 57 videos used for training and 22 videos used for visual testing and comparison. For background videos, 3 videos were taken from Background Matting and 10 videos were collected from Youtube using youtube-dl library.

### 3.4.3 Divergence loss

Since Pose2Seg segmentation gives a rough approximation of the mask, it was decided to introduce a new *Divergence loss* for network to approximately follow Pose2Seg masks so that output masks don't have any parts far from the segmentation mask given by Pose2Seg.

Loss penalizes any parts of output segmentation outside of dilated input segmentation mask. Input segmentation mask $S$ is dilated, blurred, inverted and, finally, applied to alpha matte $\alpha$ to select all regions of $\alpha$ that happen to be far outside the segmentation mask $S$. Minimized loss is:

$$||\alpha * (1 - D)||_1$$

where $\alpha$ is output alpha matte and $D$ is input segmentation mask, firstly dilated with 5 to 10 steps with kernel size 5 for resolution 512x512 and then blurred with Gaussian kernel ($\sigma = 10$). Example of loss application is demonstrated in fig. 7.

Supposedly, this loss may remove not only artifacts appearing far from segmentation contours, but also partially solves the problem of human shadow segmentation. Shadows are often included into output segmentation mask as a shadow cast by a person makes a surface underneath it darker, therefore it is considered to differ from the original background and included into the segmentation. Although, during training the network seems to exclude additional objects the person is holding or wears from the mask. For instance, if a person has a big hat or holds a guitar, it may be excluded from the segmentation. Thus, such loss must be applied only for certain use-cases where no such circumstances are present.

### 3.4.4 Training details

Network was trained similarly to [3]: learning rate of $1e^{-4}$ for the generator and $1e^{-5}$ for discriminator were used, weights were updated using Adam optimizer. Unlike in Background Matting, batch-size was set to 4 instead of 8. Same way as in Background Matting work, data was resized and cropped to 512x512 around segmentation mask. The network is implemented in PyTorch and trained for 24 hours on 2 Tesla-V100 GPUs with 16 GB memory. Random seed was fixed and data was split into ”train” and ”test” parts before training and fixed.

# 4 Experimental evaluation

This chapter compares the developed method with other approaches both qualitatively and quantitatively.

## 4.1 Comparison

A subset of videos from CDnet 2014 [6], which contain only people, was used for evaluation with other algorithms, since it is the most popular dataset for background subtraction task according to survey of video datasets for such task [21]. A total of 8 videos were taken for the evaluation out of categories ”Baseline”, ”Bad Weather” and ”Shadow”. Comparison is done with binarized masks, although our method gives ”soft” masks, i.e. with smooth edges. Some information has to be lost by thresholding segmentations.

Results of other top-performing methods were taken from the website of CDnet 2014[16] with evaluation results. Comparison of other methods with results of this work can be seen in Table 2.

SemanticBGS is the best unsupervised background subtraction method. Pose2Seg and Background Matting are human segmentation and human matting methods respectively.

In the table (2) one can see that methods developed in this work outperform methods in the comparison by all metrics. Background Matting gets results similar to Pose2Seg by F-measure due to the fact that it often adds elements not related to the person into the mask and thus has a lower precision. Worth noting, that all methods based on Background Matting have soft mask as the output, which has to be thresholded in order to perform comparison.

[16] http://changedetection.net/

| Method | Mean precision | Mean recall | Mean F-measure |
|---|---|---|---|
| SemanticBGS | 0.93 | 0.97 | 0.95 |
| Pose2Seg | 0.96 | 0.94 | 0.95 |
| Background Matting | 0.93 | 0.96 | 0.95 |
| *This work,* kpts | 0.95 | **0.98** | 0.96 |
| *This work,* div. loss | 0.96 | 0.97 | 0.97 |
| *This work,* kpts + div. loss | **0.98** | 0.97 | **0.98** |

Table 2: Comparison of the results with other methods on a subset of videos from CDnet 2014 dataset.

## 4.2 Ablation study

In the table (2) a comparison of three modifications of this method are shown. Firstly, adding keypoints to Background Matting improves both precision and recall since network can utilize information about human poses. Adding divergence loss to the initial network helps it to be closely guided by the Pose2Seg segmentation. Recall doesn't improve significantly, but precision is much higher since the network doesn't generate noise outside the person as well as removes some parts of shadows from the segmentation. Finally, keypoints and divergence loss together give the best results in terms of precision and F-measure metrics. The network is guided by segmentations but at the same time has utilized information about person pose.

## 4.3 Qualitative comparison

In Fig. (8) a visual comparison of methods that leverage keypoints is presented. The samples are from youtube dance dataset. In Fig. (9) three modifications of this method can be seen. One can notice that divergence loss partly removes shadows and any objects that differ from the background, but aren't a part of the person, as in the third row. Adding keypoints along to the divergence loss allows to utilize information about human body parts locations and fill the blanks in segmentations that appear otherwise: can be noticed in the first, third, fifth and sixth rows of (c) and (d) columns.

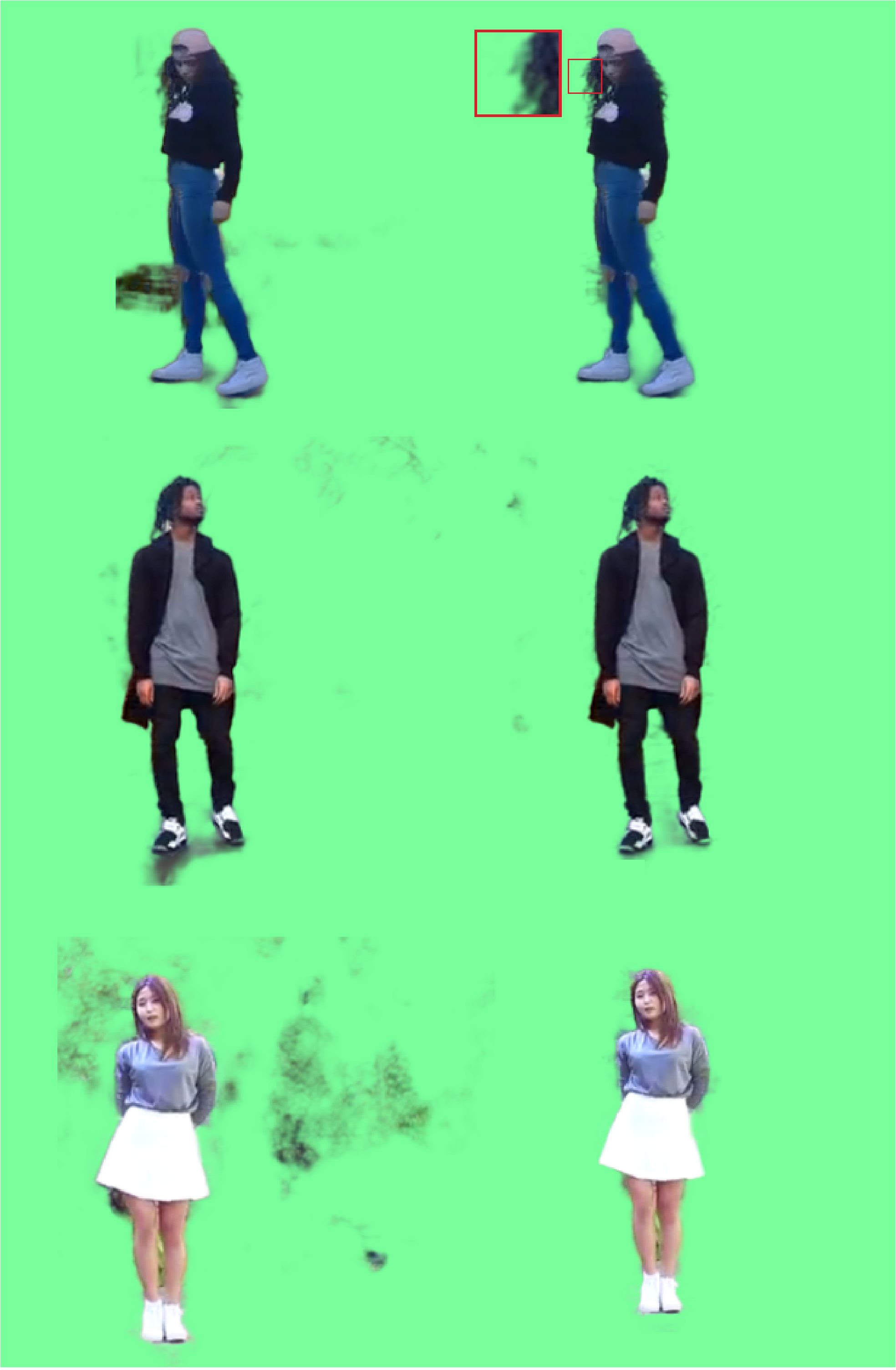

Figure 8: Comparison of methods that leverage keypoints. Left column is samples of algorithm without divergence loss, right column is samples with divergence loss.

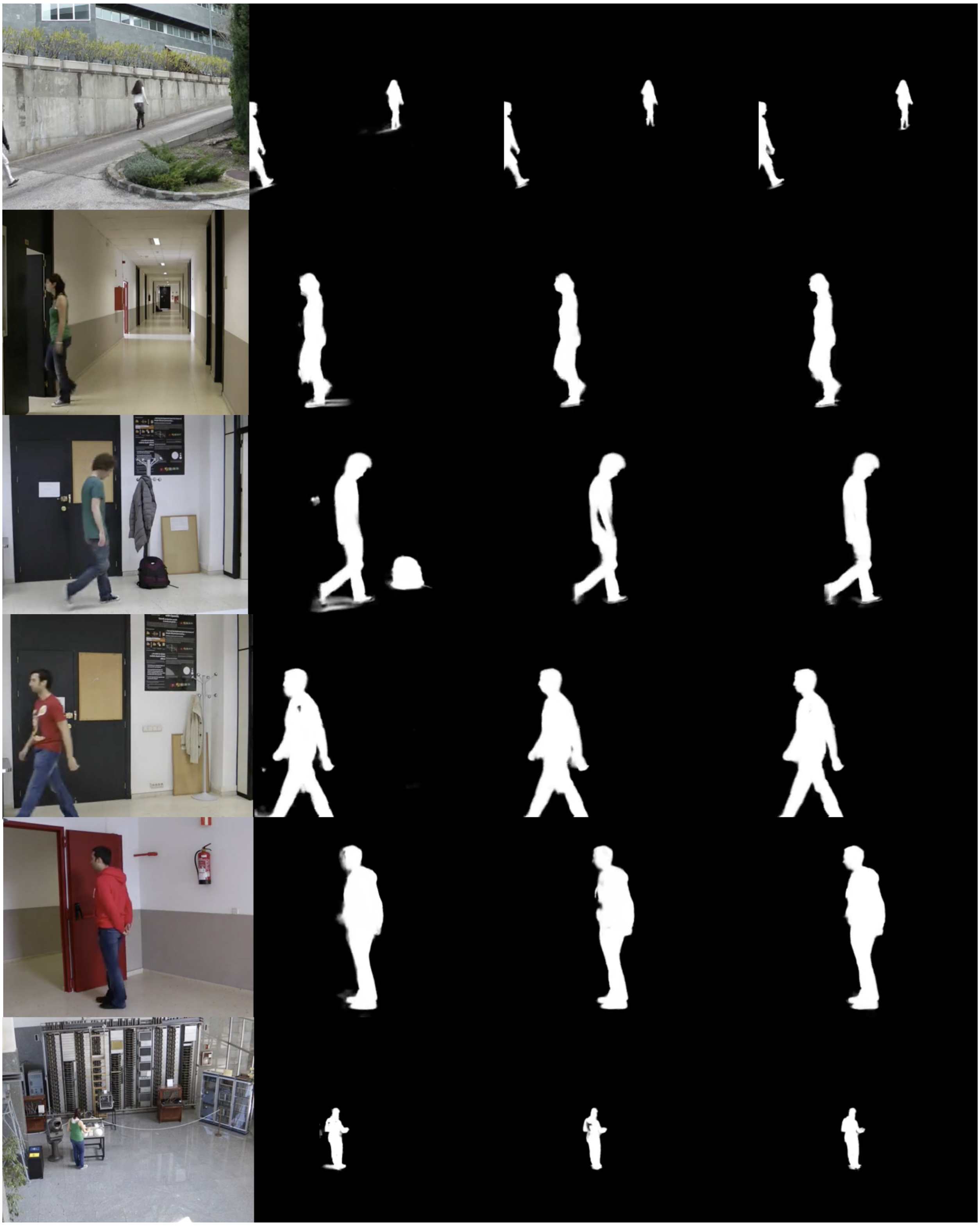

Figure 9: Comparison of method modifications on LASIESTA. From left to right: (a) source image; (b) only keypoints are added; (c) only divergence loss is added; (d) both keypoints and divergence loss are added

# 5 Web-service

In this chapter a web service task statement and implementation are discussed. Firstly, problems and prerequisites that have lead to the task of making a web service are considered, after which implementation details follow.

## 5.1 Task statement

The algorithm shows it's best performance when running on GPU, but it is a rare case when one has a powerful enough GPU available on a personal computer. Additionally, installing software on user's PC may be a complicated and unnecessary process that will increase the threshold for using the tool for segmentation. The most common case is that GPU is located on a remote server, which one could access only using SSH or similar technology. The main drawback is that there is no graphical interface, which makes it complicated to load videos and run segmentation pipelines. It would be preferable to have such an interface as it is far more convenient than loading videos and running scripts manually, especially if the system is used by users not closely familiar with the technical process like filming companies or by data science companies, who might run this service internally.

To conclude, it is needed to develop a simple web interface allowing to upload video files and download results provided by segmentation algorithm. This system is intended to be deployed on a server with GPU hardware, providing a user interface and handling requests for video processing.

## 5.2 Implementation

Web form on client-side provides an interface for uploading video files to be processed and later downloading newly generated video files. There is also an option to upload multiple videos in the form of archive. Server, which is implemented in Python, handles requests from the client-side: downloads video files, performs segmentation process and uploads them so that the user is able to download them from client-side.

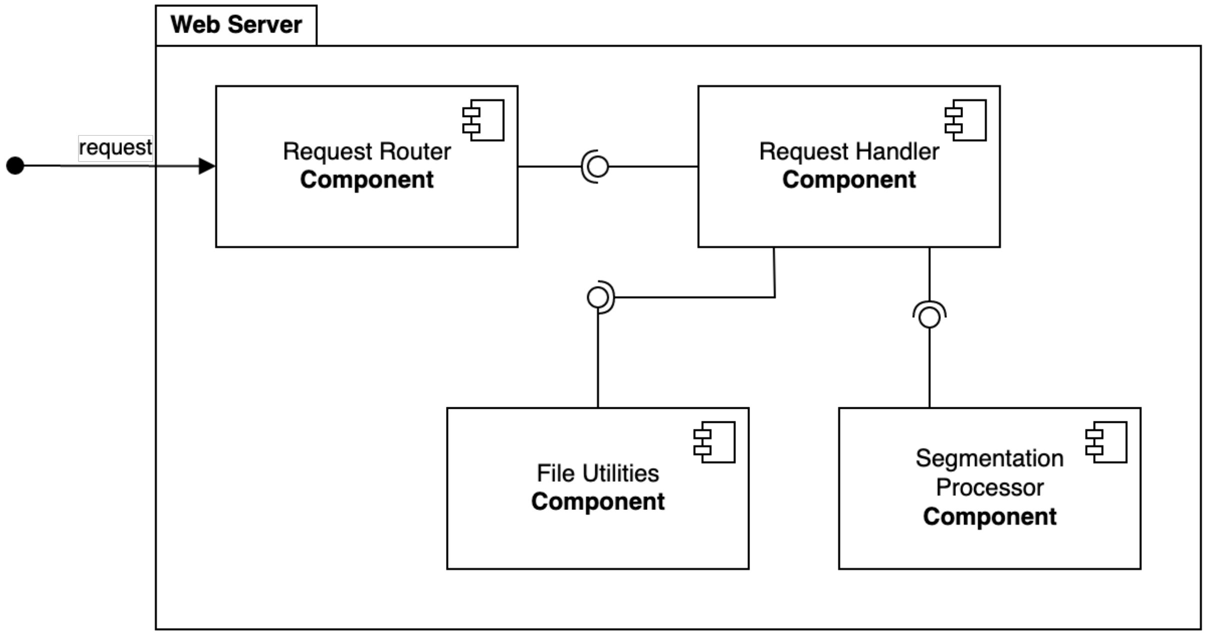

Figure 10: Server components UML-diagram

Fig. (10) represents a high-level overview of server components. Request routing is carried out by a framework, while request handling and other components are implemented such that user data is received and prepared for the segmentation pipeline, and, after being processed, all the results are gathered and returned to user. File uploading on the client side is implemented in the form of asynchronous web request to prevent user interface freezing while the data is being processed on the server. The whole system is intended to be deployed and ran inside the docker container, as the system requires a number of dependencies, and a special environment. Proper and relevant docker container, once assembled and configured, allows to deploy the system seamlessly, making distribution procedure simpler.

# Conclusion

Following results were achieved in this work.

1. A survey of the field was conducted. Both human segmentation approaches and background subtraction approaches were thoroughly examined. Suitable components of the system were chosen.
2. Pipeline for human segmentation from a video with static background was developed, consisting of a few different modules, including pose keypoint detector, segmentation module, background estimation module and mask refinement. Dataset for training was collected and processed. Existing network for refining masks was modified and trained.
3. Experimental evaluation and comparison of the developed method with SemanticBGS background subtraction method, Pose2Seg human segmentation method and original network Background Matting. Ablation study was done on the added components of the network. Qualitative visual comparison and analysis were performed.
4. Web-service was designed and developed in order to allow simpler usage of such system. Server that processes requests from the client-side and performs the segmentation, was packed into a docker container.

As a result of this work high-quality human segmentation can be utilized for tasks such as background replacement, dataset generation and others with a low barrier to entry. The system has to be deployed on a machine with an available GPU and then it's accessible for usage via web-interface on a client side. The system is modular, thus each module can be improved independently, thus increasing the quality of output results.

# References

[1] BSCGAN: deep background subtraction with conditional generative adversarial networks / Mohammed Chafik Bakkay, Hatem A Rashwan, Houssam Salmane et al. // 2018 25th IEEE International Conference on Image Processing (ICIP) / IEEE. — 2018. — P. 4018–4022.

[2] Background–foreground modeling based on spatiotemporal sparse subspace clustering / Sajid Javed, Arif Mahmood, Thierry Bouwmans, Soon Ki Jung // IEEE Transactions on Image Processing. — 2017. — Vol. 26, no. 12. — P. 5840–5854.

[3] Background Matting: The World is Your Green Screen / Soumyadip Sengupta, Vivek Jayaram, Brian Curless et al. // arXiv preprint arXiv:2004.00626. — 2020.

[4] Background and foreground modeling using nonparametric kernel density estimation for visual surveillance / Ahmed Elgammal, Ramani Duraiswami, David Harwood, Larry S Davis // Proceedings of the IEEE. — 2002. — Vol. 90, no. 7. — P. 1151–1163.

[5] Bouwmans Thierry, Garcia-Garcia Belmar. Background subtraction in real applications: Challenges, current models and future directions // arXiv preprint arXiv:1901.03577. — 2019.

[6] CDnet 2014: An expanded change detection benchmark dataset / Yi Wang, Pierre-Marc Jodoin, Fatih Porikli et al. // Proceedings of the IEEE conference on computer vision and pattern recognition workshops. — 2014. — P. 387–394.

[7] Changedetection. net: A new change detection benchmark dataset / Nil Goyette, Pierre-Marc Jodoin, Fatih Porikli et al. // 2012 IEEE computer society conference on computer vision and pattern recognition workshops / IEEE. — 2012. — P. 1–8.

[8] Chen Yucheng, Tian Yingli, He Mingyi. Monocular human pose estimation: A survey of deep learning-based methods // Computer Vision and Image Understanding. — 2020. — P. 102897.

[9] Cuevas Carlos, Yáñez Eva María, García Narciso. Labeled dataset for integral evaluation of moving object detection algorithms: LASIESTA // Computer Vision and Image Understanding. — 2016. — Vol. 152. — P. 103–117.

[10] Deep High-Resolution Representation Learning for Human Pose Estimation / Ke Sun, Bin Xiao, Dong Liu, Jingdong Wang // The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2019. — June.

[11] Deep image matting / Ning Xu, Brian Price, Scott Cohen, Thomas Huang // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. — 2017. — P. 2970–2979.

[12] Deep neural network concepts for background subtraction: A systematic review and comparative evaluation / Thierry Bouwmans, Sajid Javed, Maryam Sultana, Soon Ki Jung // Neural Networks. — 2019.

[13] Encoder-decoder with atrous separable convolution for semantic image segmentation / Liang-Chieh Chen, Yukun Zhu, George Papandreou et al. // Proceedings of the European conference on computer vision (ECCV). — 2018. — P. 801–818.

[14] Extensive benchmark and survey of modeling methods for scene background initialization / Pierre-Marc Jodoin, Lucia Maddalena, Alfredo Petrosino, Yi Wang // IEEE Transactions on Image Processing. — 2017. — Vol. 26, no. 11. — P. 5244–5256.

[15] Fast deep matting for portrait animation on mobile phone / Bingke Zhu, Yingying Chen, Jinqiao Wang et al. // Proceedings of the 25th ACM international conference on Multimedia. — 2017. — P. 297–305.

[16] Fast video object segmentation with Spatio-Temporal GANs / Sergi Caelles, Albert Pumarola, Francesc Moreno-Noguer et al. // arXiv preprint arXiv:1903.12161. — 2019.

[17] Flownet 2.0: Evolution of optical flow estimation with deep networks / Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia et al. // Proceedings of the IEEE conference on computer vision and pattern recognition. — 2017. — P. 2462–2470.

[18] Han Bohyung, Davis Larry S. Density-based multifeature background subtraction with support vector machine // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2011. — Vol. 34, no. 5. — P. 1017–1023.

[19] Hu Yuan-Ting, Huang Jia-Bin, Schwing Alexander. Maskrnn: Instance level video object segmentation // Advances in Neural Information Processing Systems. — 2017. — P. 325–334.

[20] Jampani Varun, Gadde Raghudeep, Gehler Peter V. Video propagation networks // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. — 2017. — P. 451–461.

[21] Kalsotra Rudrika, Arora Sakshi. A comprehensive survey of video datasets for background subtraction // IEEE Access. — 2019. — Vol. 7. — P. 59143–59171.

[22] Kocabas Muhammed, Karagoz Salih, Akbas Emre. MultiPoseNet: Fast Multi-Person Pose Estimation using Pose Residual Network // The European Conference on Computer Vision (ECCV). — 2018. — September.

[23] Laugraud Benjamin, Piérard Sébastien, Van Droogenbroeck Marc. LaBGen-P: A pixel-level stationary background generation method based on LaBGen // 2016 23rd International Conference on Pattern Recognition (ICPR) / IEEE. — 2016. — P. 107–113.

[24] Laugraud B., Van Droogenbroeck M. Is a Memoryless Motion Detection Truly Relevant for Background Generation with LaBGen? // Advanced Concepts for Intelligent Vision Systems (ACIVS). — Lecture Notes in Computer Science. — Antwerp, Belgium : Springer, 2017. — Sep.

[25] Learning video object segmentation from static images / Federico Perazzi, Anna Khoreva, Rodrigo Benenson et al. // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. — 2017. — P. 2663–2672.

[26] Li Xiaoxiao, Change Loy Chen. Video object segmentation with joint re-identification and attention-aware mask propagation // Proceedings of the European Conference on Computer Vision (ECCV). — 2018. — P. 90–105.

[27] Lim Long Ang, Keles Hacer Yalim. Learning multi-scale features for foreground segmentation // Pattern Analysis and Applications. — 2019. — P. 1–12.

[28] Mask r-cnn / Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick // Proceedings of the IEEE international conference on computer vision. — 2017. — P. 2961–2969.

[29] Microsoft coco: Common objects in context / Tsung-Yi Lin, Michael Maire, Serge Belongie et al. // European conference on computer vision / Springer. — 2014. — P. 740–755.

[30] Motion-guided cascaded refinement network for video object segmentation / Ping Hu, Gang Wang, Xiangfei Kong et al. // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. — 2018. — P. 1400–1409.

[31] OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields / Zhe Cao, Gines Hidalgo, Tomas Simon et al. // arXiv preprint arXiv:1812.08008. — 2018.

[32] Poisson matting / Jian Sun, Jiaya Jia, Chi-Keung Tang, Heung-Yeung Shum // ACM SIGGRAPH 2004 Papers. — 2004. — P. 315–321.

[33] Pose2Seg: Detection Free Human Instance Segmentation / Song-Hai Zhang, Ruilong Li, Xin Dong et al. // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. — 2019. — P. 889–898.

[34] Representation, analysis, and recognition of 3D humans: A survey / Stefano Berretti, Mohamed Daoudi, Pavan Turaga, Anup Basu // ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM). — 2018. — Vol. 14, no. 1s. — P. 16.

[35] Rmpe: Regional multi-person pose estimation / Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, Cewu Lu // Proceedings of the IEEE International Conference on Computer Vision. — 2017. — P. 2334–2343.

[36] Segflow: Joint learning for video object segmentation and optical flow / Jingchun Cheng, Yi-Hsuan Tsai, Shengjin Wang, Ming-Hsuan Yang // Proceedings of the IEEE international conference on computer vision. — 2017. — P. 686–695.

[37] Singh Meghna, Basu Anup, Mandal Mrinal Kr. Human activity recognition based on silhouette directionality // IEEE transactions on circuits and systems for video technology. — 2008. — Vol. 18, no. 9. — P. 1280–1292.

[38] Singh Meghna, Mandal Mrinal, Basu Anup. Pose recognition using the Radon transform // 48th Midwest Symposium on Circuits and Systems, 2005. / IEEE. — 2005. — P. 1091–1094.

[39] St-Charles Pierre-Luc, Bilodeau Guillaume-Alexandre, Bergevin Robert. Subsense: A universal change detection method with local adaptive sensitivity // IEEE Transactions on Image Processing. — 2014. — Vol. 24, no. 1. — P. 359–373.

[40] Stauffer Chris, Grimson W. Eric L. Learning patterns of activity using real-time tracking // IEEE Transactions on pattern analysis and machine intelligence. — 2000. — Vol. 22, no. 8. — P. 747–757.

[41] Temporally coherent and spatially accurate video matting / Ehsan Shahrian, Brian Price, Scott Cohen, Deepu Rajan // Computer Graphics Forum / Wiley Online Library. — Vol. 33. — 2014. — P. 381–390.

[42] Tomar Suramya. Converting video formats with FFmpeg // Linux Journal. — 2006. — Vol. 2006, no. 146. — P. 10.

[43] Video Object Segmentation and Tracking: A Survey / Rui Yao, Guosheng Lin, Shixiong Xia et al. // arXiv preprint arXiv:1904.09172. — 2019.

[44] Video matting of complex scenes / Yung-Yu Chuang, Aseem Agarwala, Brian Curless et al. // Proceedings of the 29th annual conference on Computer graphics and interactive techniques. — 2002. — P. 243–248.

[45] Xu Jiaxin, Wang Rui, Rakheja Vaibhav. Literature Review: Human Segmentation with Static Camera // arXiv preprint arXiv:1910.12945. — 2019.

[46] Zhang Tairan, Lang Congyan, Xing Junliang. Realtime Human Segmentation in Video // MultiMedia Modeling / Ed. by Ioannis Kompatsiaris, Benoit Huet, Vasileios Mezaris et al. — Cham : Springer International Publishing, 2019. — P. 206–217.

[47] A bayesian approach to digital matting / Yung-Yu Chuang, Brian Curless, David H Salesin, Richard Szeliski // Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001 / IEEE. — Vol. 2. — 2001. — P. II–II.

[48] The cityscapes dataset for semantic urban scene understanding / Marius Cordts, Mohamed Omran, Sebastian Ramos et al. // Proceedings

of the IEEE conference on computer vision and pattern recognition. — 2016. — P. 3213–3223.

[49] A survey of machine learning for big data processing / Junfei Qiu, Qihui Wu, Guoru Ding et al. // EURASIP Journal on Advances in Signal Processing. — 2016. — Vol. 2016, no. 1. — P. 67.