ARTHuS: Adaptive Real-Time Human Segmentation in Sports through Online Distillation

A. Cioppa[1,*], A. Deliège[1,*], M. Istasse[2], C. De Vleeschouwer[2] and M. Van Droogenbroeck[1]
[1]University of Liège, [2]University of Louvain, Belgium
[*]These authors contributed equally
{anthony.cioppa,adrien.deliege}@uliege.be

Abstract

Semantic segmentation can be regarded as a useful tool for global scene understanding in many areas, including sports, but has inherent difficulties, such as the need for pixel-wise annotated training data and the absence of well-performing real-time universal algorithms. To alleviate these issues, we sacrifice universality by developing a general method, named ARTHuS, that produces adaptive real-time match-specific networks for human segmentation in sports videos, without requiring any manual annotation. This is done by an online knowledge distillation process, in which a fast student network is trained to mimic the output of an existing slow but effective universal teacher network, while being periodically updated to adjust to the latest play conditions. As a result, ARTHuS allows to build highly effective real-time human segmentation networks that evolve through the match and that sometimes outperform their teacher. The usefulness of producing adaptive match-specific networks and their excellent performances are demonstrated quantitatively and qualitatively for soccer and basketball matches.

Code and video available in supplementary material.

1. Introduction

In computer vision, the task of semantic image segmentation, which consists in assigning a label to each pixel of an image, provides rich information upon which an educated understanding of the whole content of the image can be drawn [8, 14]. Regarding sports videos, semantic segmentation could be on the basis of automatic systems for *e.g.* tactics analysis, players interaction, event classification [5, 9], among numerous applications of computer vision in sports [15, 24]. However, despite being a valuable tool, semantic segmentation comes with various difficulties, which makes it an unsolved problem in the literature.

Annotation. The first challenge is the annotation issue. Semantic segmentation is generally learned as a supervised task, hence requiring ground-truth pixel-wise annotations, a process that is too time-consuming to be handled manually in every new specific context, as evidenced by the absence of any such annotated dataset in sports. To counterbalance the lack of annotated data, some authors generate synthetic training images, as in [16, 20, 22], but the quality of the data generated is often difficult to assess [12]. Another possibility to circumvent the problem of annotations is to use transfer learning strategies, that is, models that have been trained on annotated datasets are reused in a novel environment, in which no (or few) annotation is performed [17, 26]. Nevertheless, current benchmark datasets do not cover every situation where semantic segmentation might be useful, which makes transfer learning effective only when the target domain is close to the source domain; the performances can rapidly decrease otherwise. A compromise can be found by using the transfer learning procedure on a subset of selected classes as in [1], such as humans in sports scenes, in order to have a partial semantic segmentation of the image.

Speed vs performance. Then comes the trade-off between speed and performance. For instance, on the Cityscapes dataset [6], the current best algorithms [4, 32] are rather slow, while the real-time ones [19, 27, 30, 31] are not as good. Given that this dataset is meant to serve for the autonomous vehicles industry, it is essential that both performance-based and speed-based criteria are met simultaneously, which is not the case at the moment. These two aspects can also be required in sports video analysis to provide real-time accurate information about the ongoing match. A solution to benefit from the performances of a slow model (which can be designed as an ensemble of other models) and from the speed of a fast model is to perform a knowledge distillation from the slow one into the fast one [3, 11, 21, 28]. The slow accurate network has the role of a teacher, which is used as is to facilitate the training of the fast network, which has the role of a student that has to imitate its teacher's behavior on a same input dataset. After the training process, the student is supposed to be capable of real-time inference while showing good performances. In the case of semantic segmentation, this can alleviate the annotation problem for training the student if the teacher is considered reliable enough to provide approximations of unavailable ground-truth segmentation masks.

Generalization. A last problem can be the lack of generalizability of the models, whose origin is at least twofold in sports video analysis: inter-sport variability, and intra-sport variability. It is currently too ambitious to hope for a universal system that can perform accurate semantic segmentation on any sports video, which underlines the need for developing sports-specific models. Besides, even within videos from a similar view of a single sport, some play conditions may change from one match to the next, such as the teams and the color of their outfits, the advertisements, the field, and some may even change during a match, such as weather conditions in the case of outdoor events. Fast algorithms can be less robust to such variations, which might make them non-reusable from one match to the next. Rather than trying to unify all these conditions within a same network, it might be more appropriate to (re)train a scene-specific network for every match in order to adjust to the conditions of that match, in the same spirit as [2, 18]. This is the motivation behind online learning (*e.g.* [23] and references therein), in which the model is continuously updated thanks to the availability of new information.

Our contributions. In this work, we propose a novel method, named ARTHuS, to resolve at once the three issues presented above for human segmentation during live sports events. For a given match, ARTHuS produces an excellent adaptive real-time human segmentation network that evolves during the match, without having to manually annotate a single frame. This is achieved through an online distillation of a slow well-performing teacher network into a fast student network capable of real-time inference, which thus becomes match-specific. After describing the method, we evaluate the performances of the networks produced by ARTHuS quantitatively and qualitatively. We demonstrate the superiority of adaptive match-specific networks over fixed sports-specific ones. Finally, we show that the students may outperform their teacher, and we compare the performances obtained with two student architectures.

2. Method

The core problem addressed in this paper is the issue of performing an excellent real-time human segmentation in sports videos. The ideal solution would be to develop an algorithm which is well-performing, fast, and universal so that it can be used to analyze every new match, regardless of the sport. However, this last aspect is out of reach at the moment. Consequently, in order to ensure performance and speed, we sacrifice the generalizability requirement. In fact, we target the opposite: we design a method that produces match-specific networks running accurately and in real-time on the match that they are meant to analyze. This is achieved by our novel online distillation process.

Elements borrowed from usual knowledge distillation. The first step consists in choosing a "universal" trained teacher network $\mathcal{T}$ and a student network $\mathcal{S}$, possibly untrained. The teacher $\mathcal{T}$ has to be as efficient as possible for our segmentation task, even if it means that its speed has been sacrificed for the sake of performance and universality. The student $\mathcal{S}$ has to be capable to segment at least 25 frames per second to ensure real-time inference. Given an unlabeled set of images $\mathcal{X}$, the (offline) knowledge distillation from $\mathcal{T}$ into $\mathcal{S}$ on $\mathcal{X}$ can be divided into two parts:

1. Compute $\mathcal{T}(\mathcal{X})$ by feeding every image of $\mathcal{X}$ into $\mathcal{T}$ to obtain approximations of ground-truth segmentation masks.

2. Learn $\mathcal{S}$ by supervised training with the dataset $\mathcal{D} = (\mathcal{X}, \mathcal{T}(\mathcal{X}))$.

At the end of the training process, the student network $\mathcal{S}$ has learned to mimic the behavior of $\mathcal{T}$ on $\mathcal{X}$ and hopefully has become good enough to serve as a real-time segmentation network for new unseen images.

Our online knowledge distillation method, ARTHuS. In the case of sports events, many factors may change from one match to the next or even within a match. Therefore, there is no guarantee that $\mathcal{S}$, obtained by offline distillation, is able to generalize properly in novel circumstances, which motivates the idea to train a new student adaptively during every match. This leads us to propose the online distillation strategy described hereafter.

Our approach involves the following components:

1. A trained teacher $\mathcal{T}$ which remains fixed throughout the process and produces new approximations of ground-truth segmentation masks on the fly.

2. A student $\mathcal{S}^{\text{seg}}$ that performs a real-time segmentation of all the frames of the video stream. It adapts to the match as its weights are periodically updated; its k-th instance is noted $\mathcal{S}_k^{\text{seg}}$, where $\mathcal{S}_0^{\text{seg}}$is its initial instance, used to segment the first frames of the stream.

3. A training dataset $\mathcal{D}$ that is updated during the match and whose k-th instance is $\mathcal{D}_k = (\mathcal{X}_k, \mathcal{T}(\mathcal{X}_k))$.

4. A duplicate of $\mathcal{S}_0^{\text{seg}}$, denoted $\mathcal{S}^{\text{train}}$, which is trained continuously during the match with the successive instances $\mathcal{D}_k$ of $\mathcal{D}$.

Our online distillation process can be explained in two recursive steps. Firstly, $\mathcal{S}^{\text{train}}$ starts training with $\mathcal{D}_k$ (*i.e.* $\mathcal{T}$ is distilled into $\mathcal{S}^{\text{train}}$ on $\mathcal{D}_k$) while $\mathcal{S}^{\text{seg}}_{k-1}$ is used to segment all the incoming frames and $\mathcal{T}(\mathcal{I})$ is computed for some subset $\mathcal{I}$ of these incoming frames. Secondly, after a predefined number of training epochs of $\mathcal{S}^{\text{train}}$ on $\mathcal{D}_k$, the weights of $\mathcal{S}^{\text{train}}$ are copied into $\mathcal{S}^{\text{seg}}_{k-1}$, which is thus updated into $\mathcal{S}^{\text{seg}}_{k}$, and $\mathcal{D}_k$ is updated into $\mathcal{D}_{k+1}$ as the newly available pairs $(\mathcal{I}, \mathcal{T}(\mathcal{I}))$ replace as many existing pairs of $\mathcal{D}_k$. After these updates, $\mathcal{D}_{k+1}$ is available, and $\mathcal{S}^{\text{train}}$ resumes its training but with $\mathcal{D}_{k+1}$, while the rest of the process follows. This way, $\mathcal{S}^{\text{seg}}$ is a real-time segmentation network which is constantly adjusted with respect to the latest play conditions and thus becomes match-specific. The method is summarized in Algorithm 1.

Algorithm 1 The proposed online distillation algorithm.

```
Choose $\mathcal{T}$, initialize $\mathcal{S}^{\text{seg}}_0$ and $\mathcal{S}^{\text{train}}$, collect $\mathcal{D}_1$
while incoming video stream do
    while $\mathcal{S}^{train}$ trains with $\mathcal{D}_k$ do
        Segment all incoming frames with $\mathcal{S}^{\text{seg}}_{k-1}$
        Compute $\mathcal{T}(\mathcal{I})$ for some incoming frames $\mathcal{I}$
    end
    $\mathcal{S}^{\text{seg}}_{k-1}$ becomes $\mathcal{S}^{\text{seg}}_{k}$ by copying weights of $\mathcal{S}^{\text{train}}$ into $\mathcal{S}^{\text{seg}}_{k}$
    $\mathcal{D}_k$ becomes $\mathcal{D}_{k+1}$ by replacing some data with $(\mathcal{I}, \mathcal{T}(\mathcal{I}))$
    Increment $k$ by 1
end
```

As we focus on humans, through its successive instances $\mathcal{S}^{\text{seg}}_{k}$ $(k = 0, 1, \ldots)$, $\mathcal{S}^{\text{seg}}$ can be seen as an adaptive real-time human segmentation network produced by our method, which we name ARTHuS, illustrated in Figure 1.

ARTHuS depends on several choices, such as the networks and the update strategies. In this work, some particular choices have been made and are described below, but it is important to underline that our method is not limited to these choices. Many variants can be derived from the founding principle described above.

Specific settings for this work. *Data and hardware.* The sports videos used in this work are composed of frames with dimensions 1920×1080 pixels from the main camera (see supplementary material for details). The framerates provided for the algorithms are reported for images of these dimensions on one NVIDIA Tesla V100 GPU.

Teacher network. Our choice of a fixed well-performing universal teacher network $\mathcal{T}$ is Mask R-CNN [10], which runs at ≈ 2 fps on our images and has 33.8 million parameters. We only use its segmentation masks related to humans as we aim to segment humans in sports videos. Mask R-CNN operates in two steps: a detection of regions of interest followed by a segmentation within these regions. In order to focus on humans present on the field, after the detection step of Mask R-CNN, only the regions that intersect the field are kept. This filtering is performed using the segmentation mask of the field, which we compute as in [5] for our soccer experiments, and which is provided in a calibration file with the data for our basketball experiments. The whole process of collecting the results of Mask R-CNN and refining them to keep humans on the field produces training images for the instances of $\mathcal{D}$ at the speed of ≈ 1 fps.

Student network. We choose the architecture presented in [5], named TinyNet, for the fast student network $\mathcal{S}^{\text{seg}}$. Its inference speed is about 0.0165 seconds per image (≈ 60 fps) and the training time of its duplicate $\mathcal{S}^{\text{train}}$ is ≈ 0.08 second per image. It is a lightweight variant of PSPNet [32] with only 0.6 million parameters, which is about 100 times less than the original PSPNet.

Initialization. At the beginning of a new video stream, $\mathcal{D}_0$ is empty and the first minutes are used to collect and annotate data with $\mathcal{T}$ in order to build $\mathcal{D}_1$, the first non-empty instance of $\mathcal{D}$. During that time, $\mathcal{S}^{\text{train}}$ is on stand-by until $\mathcal{D}_1$ contains enough images to start its training. We consider that $\mathcal{D}_1$ is complete when it is composed of 200 annotated frames. Regarding $\mathcal{S}^{\text{seg}}_0$ (the first instance of $\mathcal{S}^{\text{seg}}$) that segments all the frames of the video stream during the building of $\mathcal{D}_1$, two approaches are tested: a random initialization, and a copy of a network pre-trained by usual offline distillation of $\mathcal{T}$ on six other matches of the same sport (see supplementary material), which is noted $\mathcal{S}_{\text{pretrained}}$.

Training. In our setting, each $\mathcal{D}_k$ is composed of 200 frames but $\mathcal{S}^{\text{train}}$ is actually trained on a subset of $\mathcal{D}_k$ covering the same game duration, built by selecting one frame every three frames. This subsampling is performed to speed up the training process of $\mathcal{S}^{\text{train}}$ and thus increases the frequency of the updates of $\mathcal{S}^{\text{seg}}$, which strengthens its adaptability during the match. We choose to train $\mathcal{S}^{\text{train}}$ during 1 epoch with the subsampled version of $\mathcal{D}_k$ before updating $\mathcal{S}^{\text{seg}}_{k-1}$ into $\mathcal{S}^{\text{seg}}_{k}$ and $\mathcal{D}_k$ into $\mathcal{D}_{k+1}$. It is trained one image at a time (no batches) using the Adam optimizer [13], and takes approximately $200/3 \times 0.08 = 5.3$ seconds per epoch. The weighted cross-entropy loss is used to handle the imbalance between human pixels and background pixels, whose ratio ranges from $1/50$ to $1/20$ in our images. The weighting factor of the loss is recomputed for each $\mathcal{D}_k$.

Updating $\mathcal{D}_k$. For each $k \geq 1$, the update strategy of $\mathcal{D}_k$ follows the "first in, first out" rule, that is, the oldest training images are replaced by the new ones. We choose to replace the oldest frames instead of just adding the new ones in order to ensure that $\mathcal{S}^{\text{seg}}_{k}$ is adapted to the latest match conditions and to keep the size of $\mathcal{D}_k$ constant, which allows to have an almost constant training time per epoch for $\mathcal{S}^{\text{train}}$ and thus regular updates for $\mathcal{S}^{\text{seg}}$.

3. Experiments

Our experiments are conducted on soccer and basketball videos. The performances of the networks produced by ARTHuS are assessed quantitatively as described below

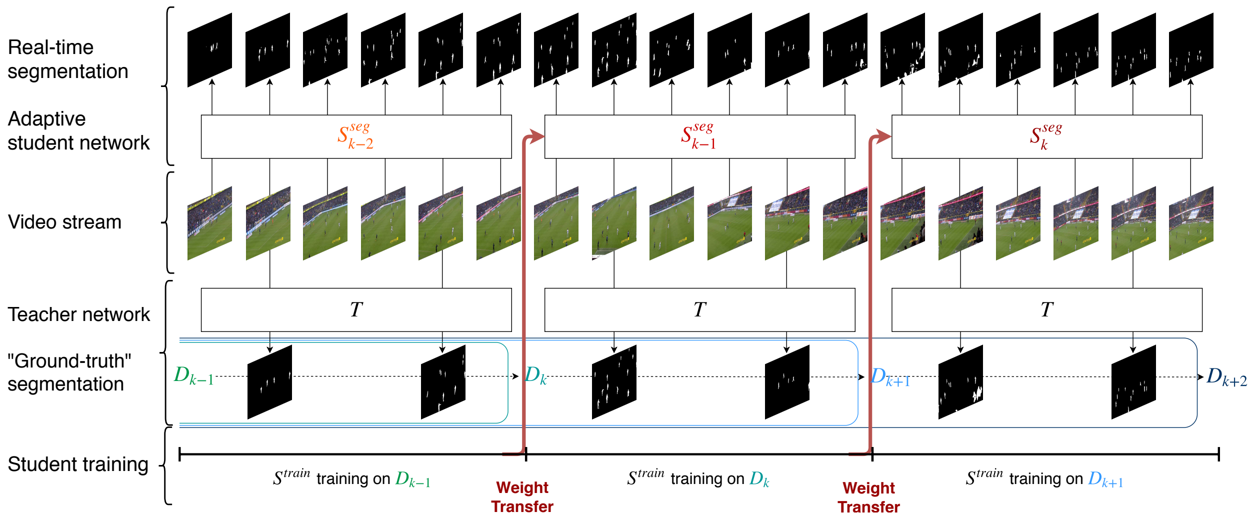

Figure 1. Illustration of our method, named ARTHuS. A real-time student segmentation network $\mathcal{S}^{\text{seg}}$ segments each frame of the video stream while its duplicate $\mathcal{S}^{\text{train}}$ continuously trains to mimic a slow but effective teacher segmentation network $\mathcal{T}$. The weights of $\mathcal{S}^{\text{train}}$ are periodically copied into $\mathcal{S}^{\text{seg}}$, which is thus consistently adapted to the latest match conditions and becomes match-specific.

and qualitatively through visual inspection of the results.

3.1. Quantitative evaluation method

As each instance $\mathcal{S}_k^{\text{seg}}(k = 0, 1, \dots)$ of $\mathcal{S}^{\text{seg}}$ produces binary masks indicating whether the pixels belong to a human (output = 1) or not (output = 0), performance metrics derived from confusion matrices can be used to represent its performances, provided that ground-truth masks are available. In such a case, the F_1 score of $\mathcal{S}_k^{\text{seg}}$ is a well-suited metric; it is computed as

$$F_1 = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}$$

where TP denotes the number of true positives (pixels correctly predicted as humans), FP the number of false positives (pixels erroneously predicted as humans), and FN the number of false negatives (pixels erroneously predicted as non-humans). However, in our case, it is difficult to obtain a large amount of ground-truth masks. For this reason, we perform the evaluation of $\mathcal{S}_k^{\text{seg}}$ in two steps. First, in Section 3.2, the evaluation is computed on frames that are annotated by $\mathcal{T}$, which we consider as sufficiently good approximations of the unavailable ground-truth masks that can be used as references. A large number of these annotated frames is available and this evaluation is meant to provide a first overview of the performances of $\mathcal{S}_k^{\text{seg}}$. Then, in Section 3.3, we manually correct the annotations of a subset of these frames to build a cleaner test dataset. This help us show that $\mathcal{S}_k^{\text{seg}}$ is mostly correct when $\mathcal{T}$ is not. It also allows to support the previous results and it attests to the reliability of that evaluation technique in our context. Finally, we use the evaluation based on $\mathcal{T}$ to compare the performances achieved by another student network in Section 3.4.

As $\mathcal{S}_k^{\text{seg}}$ can be regarded as a network trained on the frames annotated by $\mathcal{T}$ that compose $\mathcal{D}_1, ..., \mathcal{D}_k$, its evaluation has to be conducted a posteriori (not in real-time) on frames recorded after those present in $\mathcal{D}_k$. We choose to constitute the test set of $\mathcal{S}_k^{\text{seg}}$ with the N frames annotated by $\mathcal{T}$ following those of $\mathcal{D}_k$ and used to build the next instances of $\mathcal{D}$. In this work, we set $N = 300$, which spans the next five minutes of video given the framerate of $\mathcal{T}$. This way, computing the F_1 score of each $\mathcal{S}_k^{\text{seg}}$ on its test set gives the temporal evolution of the performances of $\mathcal{S}^{\text{seg}}$.

In order to handle the possible uncertainty of $\mathcal{T}$ at the borders of the humans to segment, which represents the intrinsic difficulty to perform pixel-wise annotations, some margins are drawn outside and inside these borders, whose pixels are excluded from the computation of F_1 scores. Technically speaking, these margins are computed as the Beucher gradient of the masks with a centered 7×7 structuring element, and correspond to the set difference between the morphological dilation and erosion. This practice is common in domains such as background subtraction [25], which is close to our problem in terms of evaluation of performances. This is illustrated in Figure 2.

3.2. Results

We assess the performances of the networks produced by ARTHuS on two test matches: one for soccer, one for basketball. The soccer match is the 2013 Belgian Jupiler Pro League match between FC Bruges and Anderlecht. The basketball match is the 2019 French Jeep Elite League match between Cholet and Boulazac. We chose these two matches because they both contain one unusual event out of actual game time (involving mascots), to demonstrate that

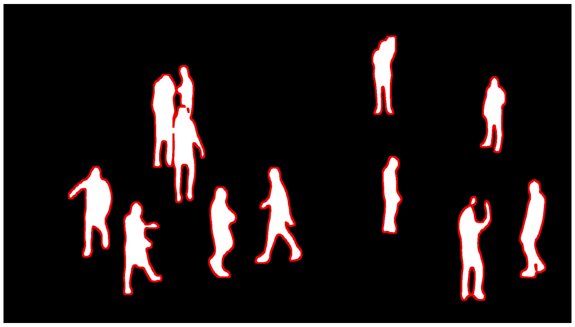

Figure 2. The Beucher gradients of the masks produced by $\mathcal{T}$ define thin margins (in red) whose pixels are excluded from the quantitative evaluation process in order to reduce the impact of the lack of accuracy of $\mathcal{T}$ at the borders of the masks on the evaluation.

our method can quickly recover from perturbations (more details are provided in supplementary material).

For each of them, we test the two strategies mentioned in Section 2 for the initialization of $\mathcal{S}_0^{\text{seg}}$: random initialization for training it online from scratch, and training it online from a sports-specific version that has been pre-trained by regular offline distillation of $\mathcal{T}$ on six other matches of the same sport, which we note $\mathcal{S}_{\text{pretrained}}$. We also compare these two approaches with the performances of $\mathcal{S}_{\text{pretrained}}$ when it is kept as is throughout the test, in order to assess its generalization capabilities and to illustrate the interest of producing adaptive match-specific networks.

The evolution of the F_1 score (with respect to the masks provided by $\mathcal{T}$) of each experiment can be found in Figure 3, where the unusual game event is marked out. It can be inferred that ARTHuS works well in practice, as indicated by the high level of performance achieved by the networks produced, regardless of the initialization strategy or the sport. In particular, even though the networks $\mathcal{S}_{\text{pretrained}}$ already have good generalization skills, the networks that are trained adaptively to become match-specific always achieve better performances after a few minutes of match. Even the networks trained from scratch with $\mathcal{S}_0^{\text{seg}}$ randomly initialized, hence without any prior knowledge on what a human on a soccer or basketball field is, eventually outperform $\mathcal{S}_{\text{pretrained}}$ and come close to those that are retrained from $\mathcal{S}_{\text{pretrained}}$. Furthermore, the networks trained online quickly recover to excellent performances after the unusual game event. Overall, the best performances are obtained with the adaptive networks that are initialized as $\mathcal{S}_{\text{pretrained}}$. These observations validate the effectiveness of the method and strengthen our point that producing adaptive match-specific networks leads to better results than using fixed sports-specific networks.

The need for match-specific networks is reinforced by the following elements. Regarding the soccer experiment, $\mathcal{S}_{\text{pretrained}}$ has been trained on matches from the UEFA Euro 2016. When we tested $\mathcal{S}_{\text{pretrained}}$ on another match from that competition, a smaller gain in performance was noted when retraining it online. This was presumably because the advertisements, camera views, stadiums, and lighting conditions, were similar to those already seen by $\mathcal{S}_{\text{pretrained}}$. However, the test match evaluated in Figure 3 is taken from another competition, the Belgian Jupiler Pro League. This match is thus rather different from those used to train $\mathcal{S}_{\text{pretrained}}$, which explains the large gap in performances between $\mathcal{S}_{\text{pretrained}}$ and $\mathcal{S}^{\text{seg}}$ on that match. Regarding the basketball experiment, the matches used to train $\mathcal{S}_{\text{pretrained}}$ belong to the same competition, but the stadiums are very different from one match to another. Therefore, there is no guarantee that $\mathcal{S}_{\text{pretrained}}$ is able to generalize correctly, and our experiment confirm that it is again beneficial to re-train it online to produce a match-specific network.

From a visual perspective, some results are displayed in Figure 4 for $\mathcal{S}_{\text{pretrained}}$ and for the networks that have been retrained online with our method with $\mathcal{S}_{\text{pretrained}}$ as initialization. The differences in the performances reported in Figure 3 are backed up by Figure 4. As $\mathcal{S}_{\text{pretrained}}$ is not specific to the test match, it cannot handle some of its peculiarities. As a result, it produces more false positives, such as the lines of the new soccer field or elements of the new basketball stadium, and more false negatives, such as partially unsegmented players, which are correctly classified by the instances of $\mathcal{S}^{\text{seg}}$ in use when these frames were recorded.

3.3. Does the student outperform its teacher?

A question that arises with knowledge distillation is whether the student network outperforms its teacher, which may occur in practice [7, 21, 29]. In our case, this question is further motivated by the observation that $\mathcal{T}$ sometimes makes mistakes while $\mathcal{S}^{\text{seg}}$ does not, as illustrated in Figure 5. This qualitative inspection suggests that $\mathcal{S}^{\text{seg}}$ may indeed outperform $\mathcal{T}$, depending on the viewer's subjective expectations to consider that $\mathcal{S}^{\text{seg}}$ surpasses $\mathcal{T}$.

From a quantitative point of view, the evaluation method presented above cannot help answering that question since the output of $\mathcal{T}$ is considered to be the ground truth with respect to which the performances of the instances $\mathcal{S}_k^{\text{seg}}$ $(k = 0, 1, \ldots)$ of $\mathcal{S}^{\text{seg}}$ are necessarily inferior or, at most, equal. Besides, the curves presented in Figure 3 are slightly flawed by the mistakes of $\mathcal{T}$ and require a manual investigation to be corrected. Manually annotating all the test frames would be too time-consuming. Also, it can be noted that $\mathcal{T}$ already segments almost perfectly most of the humans present in the test videos and that manual annotations would not be better in many cases. Therefore, we propose to build a semi-manually annotated test set, in which we manually correct the segmentation masks provided by $\mathcal{T}$ by either removing non-human pixels from the masks or by adding missed human pixels in the masks. This procedure is performed for a subset of the test frames because of lim-

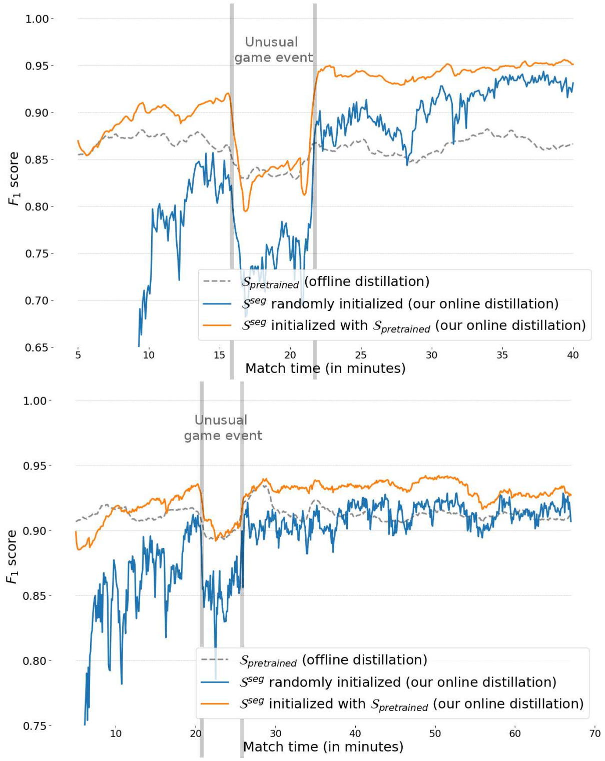

Figure 3. Evolution of the performances of several variants of our distilled models through their F_1 score computed with respect to the masks provided by $\mathcal{T}$ for the soccer (top) and basketball (bottom) test matches.

ited annotation resources. In the soccer (resp. basketball) case, 3.5% (resp. 2.5%) of the annotations have been modified, which indicates that $\mathcal{T}$ is reliable most of the time.

Considering that these corrected frames constitute the real ground truth, we can re-evaluate the performances of $\mathcal{S}_k^{\text{seg}}$ $(k = 0, 1, \ldots)$ through the match. As it can be seen in Figure 6, the F_1 score increases by a comfortable margin compared with the previous evaluation, which confirms the intuition that, when $\mathcal{T}$ is wrong, $\mathcal{S}^{\text{seg}}$ is actually mostly right. To further support this claim, we also compute the performances that $\mathcal{S}^{\text{seg}}$ would achieve if we assume that it makes no mistakes on the corrected pixels (those where $\mathcal{T}$ was considered to be wrong). This curve is also represented in Figure 6. This way, we can better quantify how good $\mathcal{S}^{\text{seg}}$ is on these new annotations, and it turns out that it is almost perfect since its adjusted performance curves are close to their upper bounds. Given that most of the annotations are still those from $\mathcal{T}$, the performance curves of $\mathcal{T}$ are unfairly higher and hence are not plotted for the sake of clarity. Let us note that the similarity between the shapes of the initial curves and the corrected curves indicates that the first approach gives a valid overview of the evolution of the performances of the networks.

Even though it is difficult to decide whether $\mathcal{S}^{\text{seg}}$ outperforms $\mathcal{T}$ or not, several qualitative and quantitative experiments show that the gap between the two is negligible and that $\mathcal{S}^{\text{seg}}$ is at least nearly as good as $\mathcal{T}$, if not slightly better.

3.4. Comparison with another student network

In order to demonstrate that the use of ARTHuS does not depend on our particular choice of student network, *i.e.* TinyNet from [5], we carry out similar experiments with another student network. For that purpose, we choose ICNet from [31], which is currently one of the best real-time segmentation networks on the Cityscapes dataset. It has been designed by some of the authors of PSPNet as a lightweight version of this architecture. ICNet has 6.7 million parameters, hence about 10 times more than TinyNet and 10 times less than PSPNet. We re-design the last layer of ICNet so that it considers only two classes, human or not human. On our hardware, its inference time is about 0.033 seconds per image ($\approx$ 30 fps) and its training time is 0.12 seconds per image.

The performances of ICNet as student network are compared with TinyNet in Figure 7 for the soccer and basketball test matches. On the one hand, when ICNet is trained online from scratch, its performances are inferior to those of TinyNet also trained online from scratch. The performance curves of ICNet increase slower than those of TinyNet, which suggests that TinyNet adapts faster to the play conditions of the ongoing match, presumably because of its reduced training time. On the other hand, when ICNet is pretrained offline through usual knowledge distillation on the same six matches as TinyNet and then retrained online, its performances are comparable to those obtained for the same experiment with TinyNet on the soccer match and are slightly higher on the basketball match, possibly because of the higher capacity of ICNet.

Consequently, ARTHuS can be used with other student architectures such as ICNet, in which case satisfying results are also obtained. Our experiments suggest that TinyNet adapts faster and better than ICNet when trained online from scratch, while ICNet shows equivalent or better performances than TinyNet when they are retrained online from a pre-trained network. However, the inference time of TinyNet is about half ICNet's, which implies that TinyNet leaves a more comfortable amount of time for potential extra real-time analyses.

4. Conclusion

We propose a novel method, named ARTHuS, that produces adative real-time human segmentation networks without requiring manual annotations. It is based on an online knowledge distillation, in which a fast student network is trained adaptively with data annotated by a slow pretrained teacher. We demonstrate the effectiveness of our method quantitatively and qualitatively on soccer and bas-

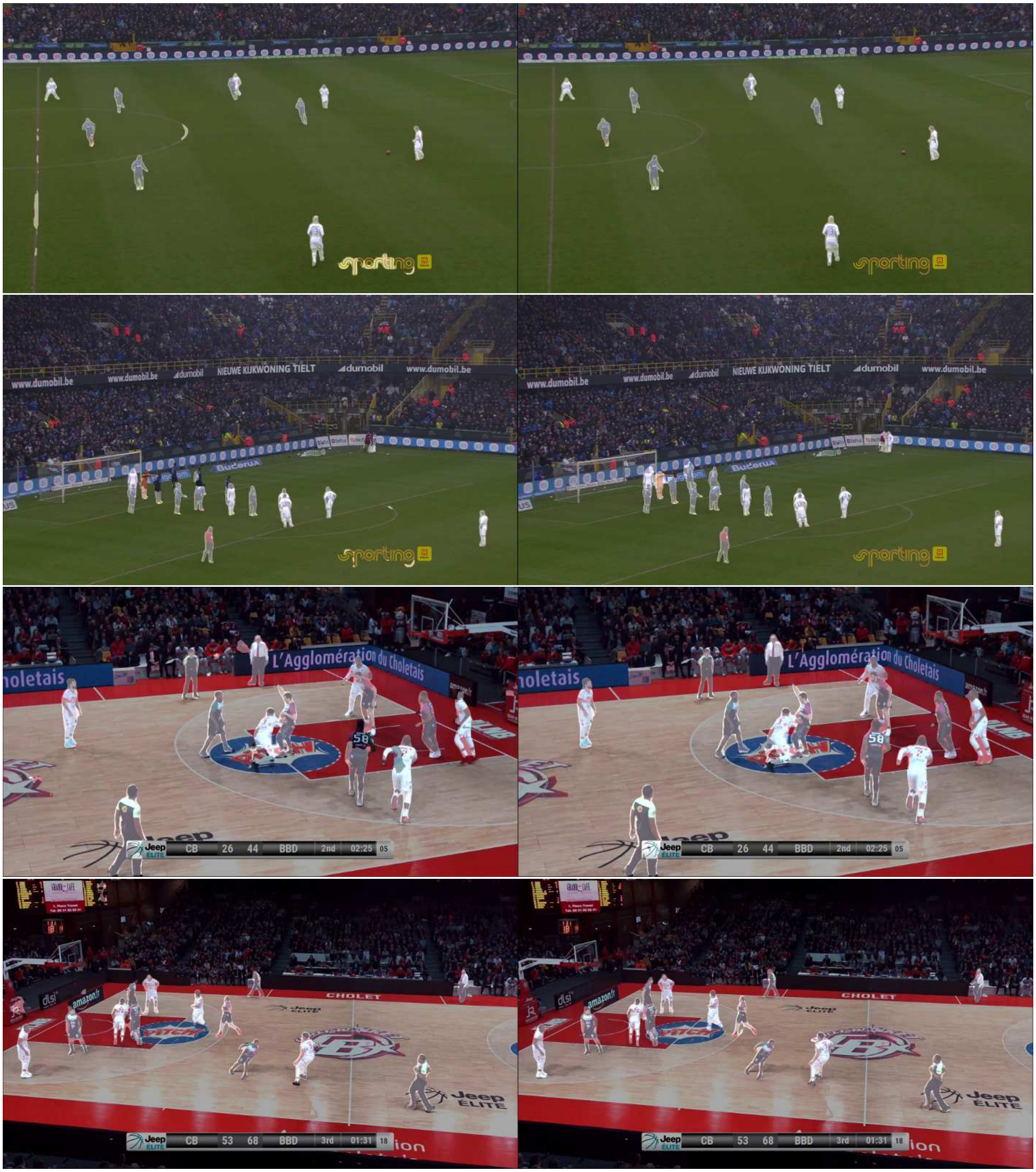
sporting
www.dumobil.be
NIEUWE KIJKWONING TIELT
dumobil
L'Agglomération du Choletais
Jeep ELITE
CB 26 44 BBD 2nd 02:25 05
CHOLET
amazon.fr
CB 53 68 BBD 3rd 01:31 18

Figure 4. Human segmentation results produced by $\mathcal{S}_{\text{pretrained}}$ (left column) and by our adaptive match-specific network $\mathcal{S}^{\text{seg}}$ produced by ARTHuS (right column) initialized with $\mathcal{S}_{\text{pretrained}}$. The effectiveness and usefulness of the adaptive network $\mathcal{S}^{\text{seg}}$ can be observed. A video showing more results is provided at https://drive.google.com/drive/folders/1FFdZYel3s8tL5YgLc6EQyZObRg2AMpDo?usp=sharing.

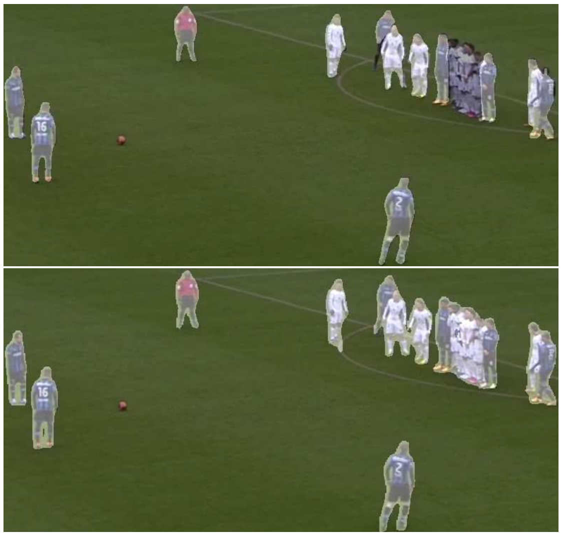

Figure 5. Example of a case where the teacher (top) does not provide a reliable output, while the instance $\mathcal{S}_k^{\text{seg}}$ (bottom) of the student network $\mathcal{S}^{\text{seg}}$ in use for this frame is actually almost flawless.

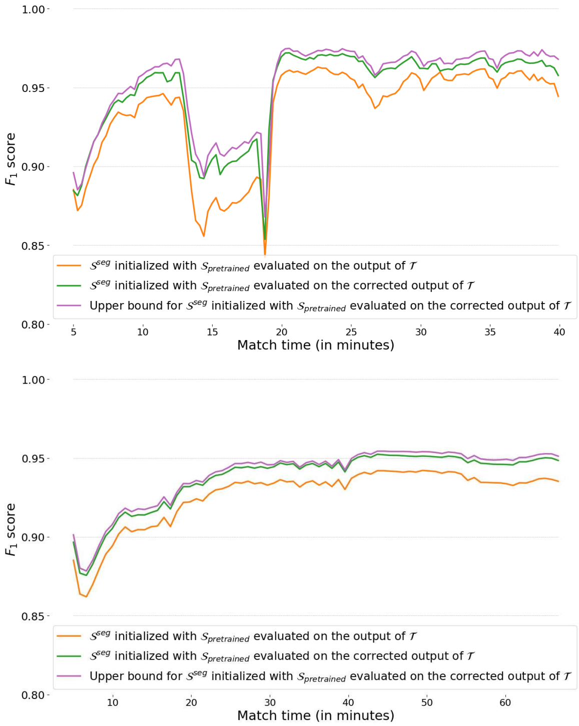

Figure 6. The previous curves (orange) are adjusted (green) by evaluating the instances $\mathcal{S}_k^{\text{seg}}$ $(k = 0, 1, \dots)$ of $\mathcal{S}^{\text{seg}}$ on the manually corrected test frames for the soccer (top) and basketball (bottom) matches. The green curves are higher, which suggests that $\mathcal{S}^{\text{seg}}$ is right when $\mathcal{T}$ is wrong. The maximum performances that $\mathcal{S}^{\text{seg}}$ would achieve if we suppose that $\mathcal{S}^{\text{seg}}$ is correct when $\mathcal{T}$ is wrong are plotted in purple. Since the green and the purple curves are close, $\mathcal{S}^{\text{seg}}$ is almost perfect on the pixels mislabeled by $\mathcal{T}$.

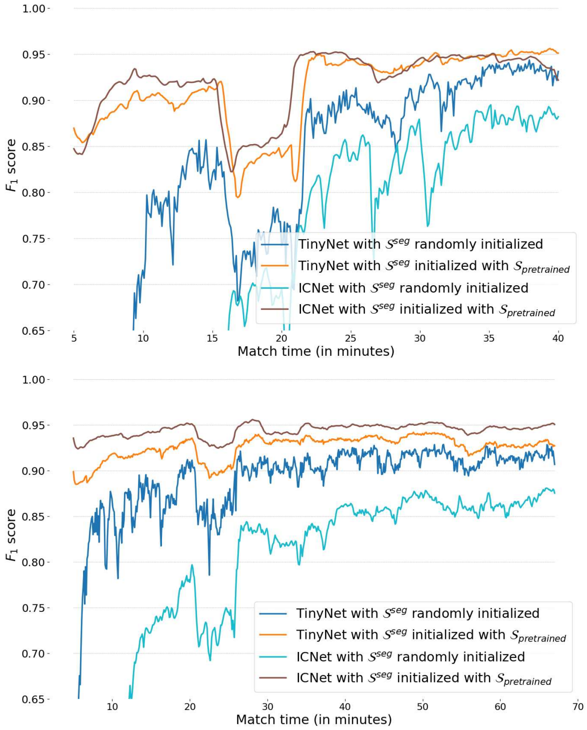

Figure 7. Comparison of the performance curves obtained with different student architectures: TinyNet [5] (*i.e.* the curves displayed previously) and ICNet [31], for the soccer (top) and basketball (bottom) test matches. $\mathcal{S}_{\text{pretrained}}$ refers to a pretrained version of the corresponding architecture, on the same set of six matches.

ketball matches. We show that match-specific networks outperform fixed pre-trained sports-specific networks, and that they eventually outperform their teacher on some occasions. We also show that ARTHuS works well with two choices of student networks, and that the architecture of TinyNet [5] might provide a better compromise than ICNet [31] between inference time, performances and adaptability.

Although ARTHuS provides promising results, there is still room for improvement, which will be investigated in future works. For instance, the update strategy of $\mathcal{D}$ can be revised in order to keep the possibility to use older frames if they are more informative than the new ones. We could also use an extra dataset that remains fixed and that would be composed of annotated frames of other matches, in order to ensure minimal generalization capabilities and enhance the robustness to possible anomalous events in the ongoing match. Besides, we can leverage the segmentation skills of our method to perform further analyses in order to develop real-time scene understanding techniques.

Acknowledgments

This research is supported by the DeepSport project of the Walloon region, Belgium. A. Cioppa is funded by the FRIA and C. De Vleeschouwer by the F.R.S.-FNRS.

References

[1] M. Braham, S. Piérard, and M. Van Droogenbroeck. Semantic background subtraction. In *IEEE Int. Conf. Image Process. (ICIP)*, pages 4552–4556, Beijing, China, Sept. 2017.

[2] M. Braham and M. Van Droogenbroeck. Deep background subtraction with scene-specific convolutional neural networks. In *IEEE Int. Conf. Syst., Signals and Image Process. (IWSSIP)*, pages 1–4, May 2016.

[3] C. Bucila, R. Caruana, and A. Niculescu-Mizil. Model compression. In *ACM Int. Conf. Knowl. Disc. and Data Mining (KDD)*, pages 535–541, Philadelphia, PA, USA, Aug. 2006.

[4] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Eur. Conf. Comput. Vision (ECCV)*, volume 11211 of *Lecture Notes Comp. Sci.*, pages 801–818. Springer, 2018.

[5] A. Cioppa, A. Deliège, and M. Van Droogenbroeck. A bottom-up approach based on semantics for the interpretation of the main camera stream in soccer games. In *Int. Workshop on Comput. Vision in Sports (CVsports), in conjunction with CVPR*, pages 1846–1855, Salt Lake City, UT, USA, June 2018.

[6] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 3213–3223, Las Vegas, NV, USA, June 2016.

[7] T. Furlanello, Z. Lipton, M. Tschannen, L. Itti, and A. Anandkumar. Born again neural networks. In *Int. Conf. Mach. Learn. (ICML)*, volume 80, pages 1607–1616, Stockholm, Sweden, July 2018.

[8] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez. A review on deep learning techniques applied to semantic segmentation. *CoRR*, abs/1704.06857, 2017.

[9] S. Giancola, M. Amine, T. Dghaily, and B. Ghanem. SoccerNet: A scalable dataset for action spotting in soccer videos. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. Workshops (CVPRW)*, pages 1711–1721, Salt Lake City, UT, USA, June 2018.

[10] K. He, G. Gkioxari, P. Dollar, and R. Girshick. Mask R-CNN. *CoRR*, abs/1703.06870, 2018.

[11] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015.

[12] M. Isogawa, D. Mikami, K. Takahashi, D. Iwai, K. Sato, and H. Kimata. Which is the better inpainted image? Training data generation without any manual operations. *Int. J. Comp. Vision*, online first, 2018.

[13] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, Dec. 2014.

[14] X. Liu, Z. Deng, and Y. Yang. Recent progress in semantic image segmentation. *Artificial Intelligence Review*, June 2018.

[15] T. Moeslund, G. Thomas, and A. Hilton. *Computer vision in sports*. Springer, 2014.

[16] F. Mueller, F. Bernard, O. Sotnychenko, D. Mehta, S. Sridhar, D. Casas, and C. Theobalt. GANerated hands for real-time 3d hand tracking from monocular rgb. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 49–59, 2018.

[17] S. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, Oct. 2010.

[18] P. Parisot and C. D. Vleeschouwer. Scene-specific classifier for effective and efficient team sport players detection from a single calibrated camera. *Comp. Vision and Image Understanding*, 159:74–88, June 2017.

[19] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello. ENet: A deep neural network architecture for real-time semantic segmentation. *CoRR*, abs/1606.02147, 2017.

[20] L. Pishchulin, A. Jain, M. Andriluka, T. Thormahlen, and B. Schiele. Articulated people detection and pose estimation: Reshaping the future. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 3178–3185, Providence, RI, USA, June 2012.

[21] A. Romero, N. Ballas, S. Kahou, A. Chassang, C. Gatta, and Y. Bengio. FitNets: Hints for thin deep nets. *CoRR*, abs/1412.6550, 2015.

[22] G. Ros, L. Sellart, J. Materzynska, D. Vázquez, and A. López. The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 3234–3243, Las Vegas, NV, USA, June 2016.

[23] D. Sahoo, Q. Pham, J. Lu, and S. Hoi. Online deep learning: Learning deep neural networks on the fly. In *Int. Joint Conf. Artificial Intell. (IJCAI)*, pages 2660–2666, Stockholm, Sweden, July 2018.

[24] G. Thomas, R. Gade, T. Moeslund, P. Carr, and A. Hilton. Computer vision for sports: current applications and research topics. *Comp. Vision and Image Understanding*, 159:3–18, June 2017.

[25] Y. Wang, P.-M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, and P. Ishwar. CDnet 2014: An expanded change detection benchmark dataset. In *IEEE*

Int. Conf. Comput. Vision and Pattern Recogn. Workshops (CVPRW), pages 393–400, Columbus, Ohio, USA, June 2014.

[26] K. Weiss, T. Khoshgoftaar, and D. Wang. A survey of transfer learning. *Journal of Big Data*, 3(1):1–9, May 2016.

[27] T. Wu, S. Tang, R. Zhang, and Y. Zhang. Cgnet: A light-weight context guided network for semantic segmentation. *CoRR*, abs/1811.08201, 2018.

[28] J. Xie, B. Shuai, J.-F. Hu, J. Lin, and W.-S. Zheng. Improving fast segmentation with teacher-student learning. In *Brit. Mach. Vision Conf. (BMVC)*, pages 1–13, Newcastle, United Kingdom, Sept. 2018.

[29] J. Yim, D. Joo, J. Bae, and J. Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 7130–7138, Honolulu, HI, USA, July 2017.

[30] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang. BiSeNet: Bilateral segmentation network for real-time semantic segmentation. In *Eur. Conf. Comput. Vision (ECCV)*, volume 11217 of *Lecture Notes Comp. Sci.*, pages 325–341. Springer, 2018.

[31] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia. ICNet for real-time semantic segmentation on high-resolution images. In *Eur. Conf. Comput. Vision (ECCV)*, volume 11207 of *Lecture Notes Comp. Sci.*, pages 418–434, 2018.

[32] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 6230–6239, Honolulu, HI, USA, July 2017.