Detecting 32 Pedestrian Attributes for Autonomous Vehicles

Taylor Mordan, Matthieu Cord, Patrick Pérez, and Alexandre Alahi

***Abstract*—Pedestrians are arguably one of the most safety-critical road users to consider for autonomous vehicles in urban areas. In this paper, we address the problem of jointly detecting pedestrians and recognizing 32 pedestrian attributes. These encompass visual appearance and behavior, and also include the forecasting of road crossing, which is a main safety concern. For this, we introduce a Multi-Task Learning (MTL) model relying on a composite field framework, which achieves both goals in an efficient way. Each field spatially locates pedestrian instances and aggregates attribute predictions over them. This formulation naturally leverages spatial context, making it well suited to low resolution scenarios such as autonomous driving. By increasing the number of attributes jointly learned, we highlight an issue related to the scales of gradients, which arises in MTL with numerous tasks. We solve it by normalizing the gradients coming from different objective functions when they join at the fork in the network architecture during the backward pass, referred to as fork-normalization. Experimental validation is performed on JAAD, a dataset providing numerous attributes for pedestrian analysis from autonomous vehicles, and shows competitive detection and attribute recognition results, as well as a more stable MTL training.**

***Index Terms*—Autonomous Vehicles, Computer Vision, Deep Learning, Multi-Task Learning, Visual Scene Understanding.**

I. Introduction

ALTHOUGH autonomous vehicles have already demonstrated successful autonomy on highways [4], [14], [31], urban areas and cities remain a challenge due to a higher degree of diversity in situations and actors. Pedestrians are arguably one of the most important categories to consider in this context, as they are more mobile and less predictable than vehicles. As part of the Vulnerable Road Users (VRUs), they have recently received more attention from the community [2], [19], [50], [54], which is essential for safety.

In the context of autonomous vehicles, lots of cues about pedestrians are available (e.g., intentions to cross the road, eye contacts with the driver, gestures for communication, descriptions of their appearances) on some datasets [46], [47]. Human attributes have proven to be effective intermediate features to combine with other tasks to improve results [60], [63], [72] or to learn more generic representations [12], [16], [33]. It opens the way for Multi-Task Learning (MTL) [9], [71] to exploit commonalities between these attributes, and to meet strict time and memory requirements of real-world autonomous vehicles by sharing computational resources between attributes.

T. Mordan and A. Alahi are with VITA, EPFL, Switzerland.
M. Cord is with LIP6, Sorbonne Université, France.
M. Cord and P. Pérez are with valeo.ai, France.
Manuscript received ...; revised ....

However, it is difficult to learn a large number of tasks simultaneously, or to learn with heterogeneous annotations, where each example is only labeled for a subset of the tasks [28]. Most previous works addressing attribute recognition in an autonomous vehicle context [47], [48], [65], [67] have only learned a limited number of attributes, missing out on some benefits of MTL. In addition, they often directly use ground-truth pedestrian bounding boxes as input, then relying on a prior pedestrian detector in real-world applications. Such results, assuming a perfect detector, are therefore optimistic [56].

In this paper, we first propose an end-to-end MTL approach, performing both detection of pedestrians and attribute recognition for all of them in a unified way based on bottom-up composite fields [30], as illustrated in Figure 1. Field formalism has proven to be effective for urban scenes, especially at low resolution, which is often the case for pedestrians imaged from a vehicle. Performing detection and attribute recognition jointly should be more computationally efficient than the two-stage pipelines implicitly implied by the previous works, and enables the evaluation of the whole perception step for a more representative understanding of the system. Secondly, we greatly increase the number of attributes to have the most informative and complete description of pedestrians, and to analyze how the model behaves when scaling MTL to numerous tasks. In this scenario, we highlight an issue with merging gradients from different tasks, and propose fork-normalization, a simple yet effective solution to handle it and improve training by normalizing gradients during back-propagation. Finally, we experimentally validate our findings on the Joint Attention in Autonomous Driving (JAAD) dataset, a dataset designed around pedestrians in an autonomous vehicle context.

II. Related Work

a) Pedestrian Analysis from Vehicles: Understanding pedestrians' behaviors around vehicles is a major milestone for safety in urban areas [40], [50], [54]. One critical interaction is road crossing [48], [56], [66], for which several large-scale datasets have recently been released to enable data-driven approaches [37], [46], [47]. Trajectory prediction is another main goal of pedestrian analysis [1], [15], [29], [46], [53], [55]. Alternative approaches jointly solve both problems by finding the short-term destinations [51], [52], or by estimating future frames and predicting target tasks from them [7], [10], [22].

b) Human Attributes: Describing humans with a set of attributes [20], [21], [34], [62] has been shown to lead to features that transfer well to other tasks [12], [33] and datasets [16]. In particular, attribute-based representations

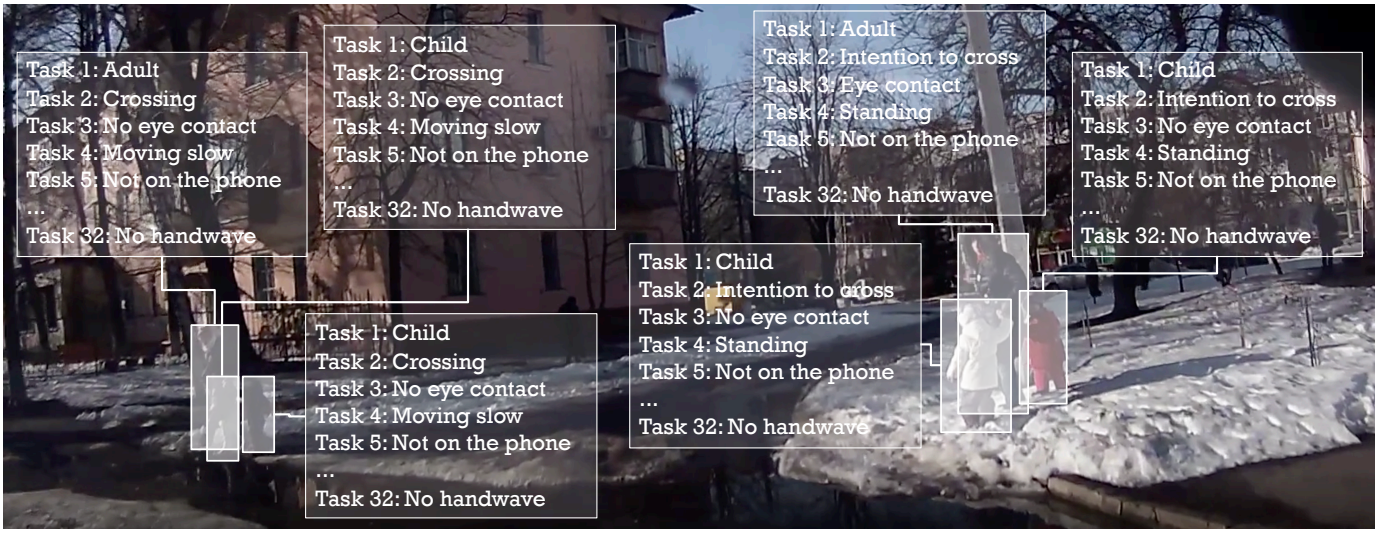


Fig. 1: **Illustration of joint pedestrian detection and attribute recognition** based on ground truth annotations from JAAD dataset [47]. We present a Multi-Task Learning (MTL) approach that jointly detects pedestrians and recognizes 32 pedestrian attributes, encompassing appearance, behavioral cues, and forecasting cues such as intention to cross the road.

are often used for person re-identification [32], [36], [58]. Regarding pedestrians, it has been shown that attributes have a direct impact on detection performances [49]. In the context of autonomous vehicles, they have been learned [44], [47], [65] and used to support other tasks, e.g., road crossing prediction [6], [27], [47], [57], [65], [67], detection [63], [72], and segmentation [60]. In particular, pose is regularly studied [17], [18], [45], [68].

c) Multi-Task Learning: The goal of MTL [9] is to share learning capacity among several tasks to benefit from transfer between them [71] and to reduce computational requirements. It has been successfully applied to a driving context [11]. However, experimentation with larger numbers of tasks has shown that boosts in performances are not systematic with enough tasks [28], and that the system has to be optimized for the specific target objectives to yield optimal transfer [5]. As an additional benefit, it has also be shown that MTL can help providing robustness against adversarial attacks [41], which is currently an important safety concern for autonomous vehicles. Most works can be divided into two categories, depending on the level at which they affect interactions among objectives. Some approaches optimize performances by learning the network structures [39], [43] or the amount of sharing between tasks [38], [42], [64], [69]. Others work with any given architecture and learn loss weights to balance tasks [13], [23], [25] or manipulate gradients to reduce negative transfer [61], [70]. These latter two particularly relate to our approach, as they all look into gradient manipulations to ensure stability during learning, from the point of view of vector angles or norms. Intuitively, the norm of the sum of gradients should be directly influenced by the angles between gradients.

This paper focuses on learning a large number of tasks in a driving context, where labels are often available for a subset of tasks only, and deals with gradient scale issues for a generic MTL architecture.

III. Joint Pedestrian Detection and Attribute Recognition

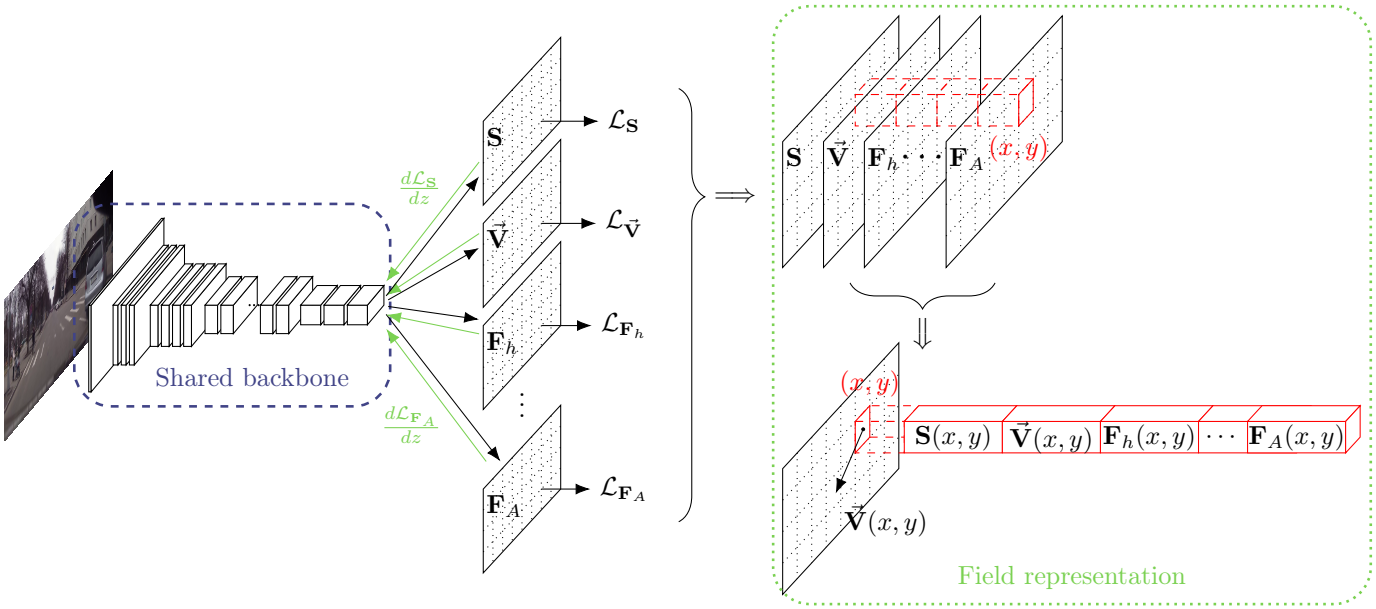
Our model is based on the common Multi-Task Learning (MTL) network architecture consisting of a single shared backbone followed by a separate predictor for each task. We rely on the composite field formalism [30], where the model's output is a set of spatial feature maps encoding both the locations and attributes of all the pedestrians in the scene. Detection and attribute recognition tasks are here combined in a unified approach, naturally lending itself to MTL [63], [72].

A. Field Formalism

For every pedestrian attribute f we want to predict (e.g., location, bounding box dimension, visual or behavioral attribute), we consider an associated field $\mathbf{F}_f$ (for clarity in the following, we will omit the subscript f when it is clear from the context), which is a spatial map on whole images and whose cells $\mathbf{F}(x, y)$ locally encode an estimation of the given attribute f based on the neighborhoods around the locations (x, y).

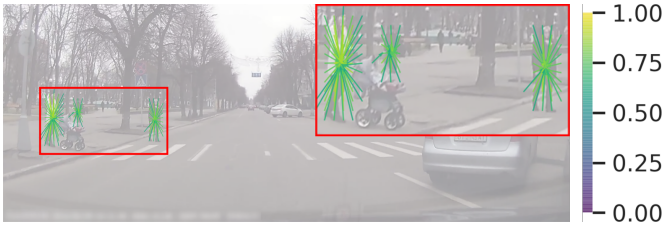
We consider two kinds of fields. Scalar fields $\mathbf{F}$ encode global attributes f about pedestrians from all their respective cells regardless of their relative locations (see Figure 2 (b) for an example). These attributes can be binary (e.g., intention to cross the road), categorical (e.g., age category), or continuous (e.g., height). In case of a categorical attribute, the field $\mathbf{F}$ has as many channels as classes for the attribute, and $\mathbf{F}(x, y)$ contains predictions for all these classes. A vectorial field $\vec{\mathbf{F}}$ points to spatial locations $\vec{f} = (x^f, y^f)$ relative to the pedestrians' positions, with vectors pointing from all their respective cells to the desired locations (see Figure 2 (c) for an example). The only vectorial attribute we have in this paper is the locations of the instances' centers, but this formulation could be used with any localized attribute, such as pose keypoints for example.

As depicted in Figure 2 (a), when taking all such fields from the model stacked together, every cell (x, y) of the output



(a) Architecture of our model

(b) Field $\mathbf{S}$



(c) Field $\vec{\mathbf{V}}$

Fig. 2: **Field-based representations.** (a) The model is composed of a backbone branching into a task-specific branch for each field. Detection of pedestrian instances is done with fields $\mathbf{S}$ and $\vec{\mathbf{V}}$, with $\mathbf{F}_h$ and $\mathbf{F}_w$ for the bounding boxes' dimensions. The recognition of other attributes is done with additional fields $\mathbf{F}_1, \ldots, \mathbf{F}_A$ for A attributes. When considering all fields together, at each cell (x, y) in the model's output, there is a local representation localizing the center of the pedestrian it belongs to, along with predictions for all attributes. During inference (not shown in the figure), all the local attributes associated with a same pedestrian are aggregated through a vote to yield a unique prediction per detection. (b-c) Example fields $\mathbf{S}$ and $\vec{\mathbf{V}}$, with top right windows zooming in on pedestrians for more details. The color code for the heatmap from $\mathbf{S}$ and the arrows from $\vec{\mathbf{V}}$ is based on the confidence scores $\text{Sigmoid}\,(\mathbf{S}(x, y))$ of the cells.

contains a complete representation around this location. This includes an estimation of the presence of a pedestrian, and in the case of a detected pedestrian, the associated location and local predictions for all attributes. When no pedestrian is detected at the location, the attribute predictions will not be used and do not affect the results.

This image-wise view based on fields is later converted into a set of instance-wise predictions with a decoding post-processing step: all cells (x, y) associated with the same pedestrians are aggregated to yield a single prediction f_p or $\vec{f}_p$ from field $\mathbf{F}$ or $\vec{\mathbf{F}}$ for detected pedestrian p (see Section III-D). These spatial formulation and inference implicitly take context into consideration and are therefore well suited to low resolution contexts [30].

B. Detection and Attribute Recognition

Pedestrian detection is achieved with multiple fields. First, field $\mathbf{S}$ (see Figure 2 (b)) estimates how likely cells (x, y) are to belong to pedestrian instances. This is done through confidence scores $\text{Sigmoid}\,(\mathbf{S}(x, y)) \in [0, 1]$: a value close to 1 indicates a pedestrian, while a value close to 0 indicates background. Then, field $\vec{\mathbf{V}}$ (see Figure 2 (c)) is used to localize the centers of the pedestrian instances, i.e., the centers of their bounding boxes. At any cell (x, y) belonging to a pedestrian, the vector $\vec{\mathbf{V}}(x, y)$ should point from the cell to the corresponding pedestrian's center. The two fields $\mathbf{S}$ and $\vec{\mathbf{V}}$ are used together to detect all pedestrians present in the images and spatially cluster cells into separate instances in a bottom-up way [25]. This does not require any hand-crafted modules that are common in detection pipelines, such as prior boxes or non-maximum suppression step, which is itself a main motivation of recent object detection methods [8], [73].

However, the fields $\mathbf{S}$ and $\vec{\mathbf{V}}$ only allow for point detection, i.e., detecting pedestrians by their central point. Regular box detection is obtained by adding two other scalar fields, for the heights and widths of the bounding boxes. Grouped together, these form the first attribute we consider for pedestrians: the bounding box attribute. Note that although the field $\mathbf{S}$ already provides a coarse segmentation of the pedestrians in the feature space, this is not accurate in the image space due to the large stride of the network. Also, getting the bounding box as a field prediction should be more robust, especially in the case of occlusion, as it is averaged over multiple cells rather than relying on the detection of cells at the edges of the pedestrians.

Once all cells are attributed to specific pedestrian instances or background, attribute recognition is straightforward. Given an attribute f (including height and width) or $\vec{f}$ (including the center of the detection) with associated field $\mathbf{F}$ or $\vec{\mathbf{F}}$, a prediction for a pedestrian instance is obtained by a vote of all values $\mathbf{F}(x,y)$ or $\vec{\mathbf{F}}(x,y)$ whose cells (x,y) have been clustered in this instance.

C. Image-wise Learning

We consider annotations for the tasks of detection and attribute recognition under the form of a bounding box for each pedestrian, along with a list of attributes. The attributes can be scalar (binary, categorical or continuous values) or vectorial (spatial coordinates). Note that pedestrians are not all annotated for all the attributes, depending on what is visible in the data, making the set of attributes available for each pedestrian different.

Learning is carried out with image-wise loss functions, independently of the number of pedestrians present in the scene. Binary and categorical fields can be learned with any classification loss function, and continuous and vectorial fields with any regression loss function. Binary targets for the detection field $\mathrm{Sigmoid}\,(\mathbf{S})$ are set to 1 for all cells (x,y) belonging to pedestrians, defined as (x,y) being within a pedestrian's bounding box, and to 0 if they correspond to background. For all other fields, there are no targets defined for background cells and no loss is back-propagated there. Regarding a pedestrian cell (x,y), all fields learn the attributes of the corresponding pedestrian. Scalar fields $\mathbf{F}$ first go through activation functions act to have the right output ranges: $\mathrm{Sigmoid}$ for binary attributes, $\mathrm{Softmax}$ for categorical attributes and $\mathrm{Identity}$ for continuous attributes. Targets are then set to the values of the corresponding pedestrian's attributes f, duplicated over all the cells within the bounding box. For vectorial fields $\vec{\mathbf{F}}$, the targets are the vectors pointing from that cell (x,y) to the locations $\vec{f}=(x^f,y^f)$ of the attributes, i.e., $\vec{f}-(x,y)$. Note that the only vectorial field used in this paper is $\vec{\mathbf{V}}$, but the setup would be the same for any other such field.

D. Instance-wise Decoding

As displayed in Figure 2 (a), the model's output is a set of fields on whole images. Predictions are obtained from this image representation in a bottom-up way, with a post-processing step not requiring training. Pedestrian detection is first done by clustering cells into a set of instances, using fields $\mathbf{S}$ and $\vec{\mathbf{V}}$. Then, attribute recognition for all instances previously found is achieved by aggregating the attribute fields over these instances.

First, we select cells (x,y) whose confidence values $\mathbf{S}(x,y)$ are greater than a given threshold γ, in order to keep likely pedestrian cells only. As in [25], the estimated centers $(x,y)+\vec{\mathbf{V}}(x,y)$ pointed at by the field $\vec{\mathbf{V}}$ from retained cells (x,y) are then clustered into a set of instances with OPTICS algorithm [3], which has the property of not needing a prior number of clusters. This yields P clusters $\mathcal{C}(1),\ldots,\mathcal{C}(P)$, i.e., P pedestrian detections, where each cluster $\mathcal{C}(p)$ contains all the cells (x,y) associated with pedestrian p.

For each detected pedestrian p, associated with cluster $\mathcal{C}(p)$, we compute global predictions for all attributes based on the cells (x,y) in the cluster. First, the confidence score s_p for the detection, used to rank detections for evaluation, is the average of scores $\mathbf{S}(x,y)$ over all cells:

$$s_p = \mathrm{Sigmoid}\left(\frac{1}{|\mathcal{C}(p)|}\sum_{(x,y)\in\mathcal{C}(p)}\mathbf{S}(x,y)\right), \tag{1}$$

where $|\mathcal{C}(p)|$ is the number of cells in the cluster. Then, all other predictions are weighted averages with weights coming from the field $\mathbf{S}$. A scalar (binary, categorical or continuous) attribute prediction f_p is given by

$$f_p = \mathrm{act}\left(\frac{\sum_{(x,y)\in\mathcal{C}(p)}\mathrm{Sigmoid}\,(\mathbf{S}(x,y))\,\mathbf{F}(x,y)}{\sum_{(x,y)\in\mathcal{C}(p)}\mathrm{Sigmoid}\,(\mathbf{S}(x,y))}\right), \tag{2}$$

where act is an activation function used to get the right output range depending on the attribute type, as defined in Section III-C. Note that for the categorical case, $\mathbf{F}(x,y)$ have values for all classes of the attributes. A vectorial attribute prediction $\vec{f}_p$ is computed by

$$\begin{aligned}\vec{f}_p &= (x_p^f, y_p^f)\\ &= \frac{\sum_{(x,y)\in\mathcal{C}(p)}\mathrm{Sigmoid}\,(\mathbf{S}(x,y))\left((x,y)+\vec{\mathbf{F}}(x,y)\right)}{\sum_{(x,y)\in\mathcal{C}(p)}\mathrm{Sigmoid}\,(\mathbf{S}(x,y))}.\end{aligned} \tag{3}$$

We use this architecture to jointly do pedestrian detection and attribute recognition for each of the detections. However, scaling to numerous attributes in a multi-task framework is notoriously difficult [28], and often leads to drops in performance with respect to the single-task models. In the next section, we study this phenomenon more in depth for our model.

IV. Scaling Learning to Multiple Attributes

We now analyse the behavior of the MTL network at the fork in the architecture, between the shared backbone and all the task-specific heads, when varying the number of tasks. We first shed light on an issue about gradient scales, and then suggest a simple solution to fix it and stabilize training.

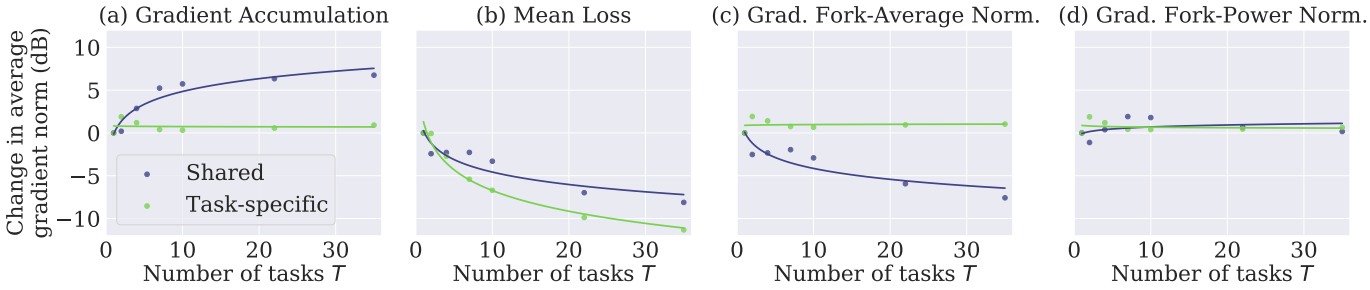


Fig. 3: **Relative change in average gradient norm** (in dB), in both the shared backbone and task-specific sub-networks, when increasing the number of tasks T learned with (a) standard gradient accumulation; (b) global mean of loss functions; (c) our proposed gradient fork-average normalization; (d) our proposed gradient fork-power normalization. Norm values are the average norms of gradients over the first epochs. Points are actual measurements and lines are linear fits in log-log spaces.

A. MTL Gradient Scale Issue

We consider the common MTL architecture for T tasks, composed of a shared backbone branching into T predictors. For our model, this corresponds to one task for each field. Each branch t is associated to the corresponding task t with loss function $\mathcal{L}_t$ and task weight λ_t. For simplicity, we assume for now that all examples are annotated for all T tasks, but we later describe a slight modification to apply when annotations are heterogeneous across examples, as is the case in this paper. As usual, the loss function $\mathcal{L}$ for the whole model is the weighted sum of all branch losses:

$$\mathcal{L} = \sum_{t=1}^{T} \lambda_t \mathcal{L}_t. \tag{4}$$

During the backward pass, for a feature z from the shared backbone, the gradient $d\mathcal{L}/dz$ can be bounded by the norms of all task-specific gradients at the same point:

$$\left\| \frac{d\mathcal{L}}{dz} \right\| \leq \sum_{t=1}^{T} \left\| \lambda_t \frac{d\mathcal{L}_t}{dz} \right\| \leq T \max_{1 \leq t \leq T} \left\| \lambda_t \frac{d\mathcal{L}_t}{dz} \right\|. \tag{5}$$

Assuming the weights λ_t are not normalized by T (e.g., by forcing them to sum to 1), which is a common case [13], [23], [25], the bound on the norm directly grows with T. Although this does not imply that the gradients will actually grow with the bound, we experimentally observe it, as shown in Figure 3 (a). This exploding gradient phenomenon in the shared backbone could then lead to unstable training with too many tasks.

A simple solution to this issue could be to average the losses rather than summing them, i.e., normalizing the weights λ_t by T in Equation (4), or, equivalently, decreasing the learning rate by a factor T. The bound on the gradient then becomes

$$\left\| \frac{d\mathcal{L}}{dz} \right\| \leq \frac{1}{T} \sum_{t=1}^{T} \left\| \lambda_t \frac{d\mathcal{L}_t}{dz} \right\| \leq \max_{1 \leq t \leq T} \left\| \lambda_t \frac{d\mathcal{L}_t}{dz} \right\|, \tag{6}$$

which is independent of T and prevents the backbone gradients to explode (assuming all loss functions are stable), as observed in Figure 3 (b). However, the effective loss weights λ_t/T now decrease with T. Since for a feature (including output) z_t in a task-specific branch t (i.e., after the fork), the loss $\mathcal{L}_t$ is the only one influenced by z_t, the gradient with respect to z_t becomes

$$\frac{d\mathcal{L}}{dz_t} = \frac{\lambda_t}{T} \frac{d\mathcal{L}_t}{dz_t}. \tag{7}$$

The norm of this gradient now decreases with T too, so that the task-specific branches learn slower with more tasks.

We interpret this as a difference in learning dynamics between the shared and task-specific sub-networks inherent to MTL, which should be more pronounced the more tasks there are. This was already hinted at in UberNet [28], which uses a different counter for each part (any task-specific predictor or the shared backbone) and updates them separately depending on the number of examples back-propagated through each of them. We here formalize the issue more clearly.

B. Fork-Normalizing MTL Gradient Back-Propagation

In order to get the best of both worlds, i.e., gradient bounds independent of the number of tasks T both in the shared backbone and the task-specific heads, we propose to modify how gradients join at the fork during the backward pass. We introduce a set of T parameters $\kappa = (\kappa_1, \ldots, \kappa_T)$ to weigh the gradients coming from different branches in the merging, which we call fork-normalization. Note that the forward pass (Equation (4)) is left unchanged, only the learning dynamic is adapted during the backward one, which now becomes:

$$\begin{cases} \dfrac{d\mathcal{L}}{dz} = \displaystyle\sum_{t=1}^{T} \kappa_t \lambda_t \frac{d\mathcal{L}_t}{dz}, & \text{for } z \text{ in backbone,} \\ \dfrac{d\mathcal{L}}{dz_t} = \lambda_t \dfrac{d\mathcal{L}_t}{dz_t}, & \text{for } z_t \text{ in branch } t. \end{cases} \tag{8}$$

There are now two sets of parameters λ_t and κ_t associated with tasks t, but they have different purposes. The first are used to balance the relative importances of tasks, to bias learning toward any of them. The weights can be chosen by cross-validation, or tuned dynamically based on the uncertainty inherent to the tasks [25] or on the relative rate at which they learn [13], [23] for example. On the other hand, the κ parameters are used to balance learning in the shared backbone relative to the task-specific branches, and we propose strategies to choose them in this paper.

Several ways are possible for choosing κ. The standard gradient accumulation formulation is obtained by setting $\kappa_t = 1$ for all t. We first propose to choose κ so that the coefficients sum to 1, i.e., $\sum_{t=1}^{T} \kappa_t = 1$. As shown in Figure 3 (c), this yields a stable bound on the backbone gradient norm as in Equation (6), while still keeping the learning of the task-specific branches unchanged and independent of T. For each example, we sample a T-tuple from a symmetric Dirichlet distribution $\kappa \sim \text{Dir}(\alpha)$ where all the T dimensions have the same concentration parameter α. We study three particular cases of fork-normalization here:

(i) random, where κ is sampled from the uniform distribution over the T-tuples ($\alpha = 1$);
(ii) sample, where a single task t is selected for back-propagation in the backbone with $\kappa_t = 1$ ($\alpha \to 0$);
(iii) average, where all $\kappa_t = 1/T$ have the same constant value ($\alpha \to +\infty$).

Although this approach yields a stable upper bound on the gradient norm (independent of T), we experimentally observe in Figure 3 (c) that the actual norm decreases in the backbone when increasing the number of tasks T, meaning that the bound on gradient norm from Equation (5) is not tight. To address this, we model the relation between the gradient norm and number of tasks as linear in log-log space, i.e., a power law. This yields new gradient weighting parameters of the form $\kappa_t = 1/T^{\beta}$, where β is a hyper-parameter of the power law. From the measurement data in the graphs from Figure 3, we obtain $\beta = 0.5$ under this modeling. Results for this gradient fork-power normalization with $\kappa_t = 1/\sqrt{T}$ are displayed in Figure 3 (d), where we observe that gradient norms in both the shared backbone and task-specific heads are now much more stable with respect to the number of tasks T.

By keeping gradients at the same scale regardless of the number of tasks, the training should be more stable, with more balance between shared backbone and task-specific branches. Hyper-parameters such as learning rate should also generalize better to the addition of tasks, making cross-validation easier to carry out. In addition, this modification does not incur any overhead. In the case of a single task ($T = 1$), all methods are completely equivalent to the standard gradient accumulation.

Until here, we assumed for simplicity that targets for all T tasks were always available. When this is not the case for a given example, i.e., it is annotated only for a subset of $\widetilde{T} < T$ tasks, we apply the same strategy with the effective number of back-propagated tasks $\widetilde{T}$ in the normalizing factor κ. For example, the fork-power normalization weights become $\kappa_t = 1/\sqrt{\widetilde{T}}$. Note that $\widetilde{T}$ can now be different across examples.

Gradient fork-averaging should be equivalent to having separate learning rates for the shared backbone and the task-specific heads, and adapting them based on the number of tasks. Adam [26] is an optimizer that maintains a different learning rate per parameter, and could therefore be able to achieve similar results automatically. However, we experimentally find that the issue is still present with Adam.

V. Experiments

A. Experimental Setup

a) Dataset and Attributes: We use JAAD dataset [47], which is centered around pedestrian analysis from vehicles. To the best of our knowledge, it is the only dataset where pedestrians are annotated both for detection and with a large number of diverse attributes. We consider the default split, composed of 40,530 images (177 videos) for training, 7,170 images (29 videos) for validation, and 27,912 images (117 videos) for testing. We extract 32 pedestrian attributes in total, which we divide in four sets of A attributes:

(a) Detection only, i.e., bounding box attribute ($A = 1$), but note that it corresponds to four task losses associated to fields $\mathbf{S}$, $\vec{\mathbf{V}}$, $\mathbf{F}_h$ and $\mathbf{F}_w$ as explained in Section III;
(b) Future intention attributes ($A = 2$):
- binary: 'road crossing intention';
- continuous: 'time-to-crossing';

(c) Current behavior attributes ($A = 10$):
- binary: 'instant road crossing', 'looking', 'walking', 'motion direction', 'back pose', 'front pose', 'left pose' and 'right pose';
- categorical: 'group size' and 'reaction';

(d) Appearance attributes ($A = 19$):
- binary: 'gender', 'backpack', 'bag at elbow', 'bag at hand', 'bag on left side', 'bag on right side', 'bag on shoulder', 'cap', 'clothes below knee', 'dark lower clothes', 'dark upper clothes', 'light lower clothes', 'light upper clothes', 'hood', 'object', 'phone', 'stroller cart' and 'sunglasses';
- categorical: 'age'.

We left out attributes about 'hand gesture', 'nod', 'baby', 'bicycle/motorcycle' and 'umbrella' because there are too few examples to learn from for these attributes. Note that all attributes are not available on all pedestrians, making the annotations heterogeneous across examples. We also do not train on pedestrians too heavily occluded. More details can be found in the reference paper [47].

In order to analyze the behavior of our model when varying the number of tasks, we train and evaluate on four increasing sets of attributes:

- (a) with $A = 1$;
- (a+b) with $A = 3$;
- (a+b+c) with $A = 13$;
- (a+b+c+d) with $A = 32$.

b) Implementation Details: Our model is based on a ResNet-50 backbone [24], with single 1×1 sub-pixel convolution layers [59] as task-specific predictors. We use pre-trained weights from PifPaf [30] since it uses a similar framework and is trained on humans specifically. The loss functions are (binary) focal cross-entropy [35] for (binary) classification tasks, and L_1 for regression ones (continuous scalar and vectorial attributes).

The networks are trained with SGD optimizer with a batch size of 4, learning rate of $5 \cdot 10^{-4}$, weight decay of $5 \cdot 10^{-4}$, momentum of 0.95, and exponential model averaging with decay constant of 10^{-3}. The number of epochs is selected based on convergence for each variant and group of attributes

TABLE I: **Comparison between multi- and single-task networks** on JAAD val set with $A = 32$ attributes. APs (%) are shown for some attributes, mAP (%) is the average of APs for all $A = 32$ attributes. TtC stands for Time-to-Crossing.

			Detection	Intention		Behavior			Appearance			All
MTL Strategy	Networks	Memory	Pedestrian	Crossing	TtC	Looking	Walking	Front Pose	Age	Gender	Phone	mAP
32 Networks	32 ResNet-18's	38.4GB	65.1	58.9	22.3	**35.7**	27.2	46.2	13.4	30.2	31.0	36.6
MTL Baseline	1 ResNet-50	**1.6GB**	66.3	56.5	21.9	34.4	28.4	45.3	21.1	32.1	32.2	36.6
PCGrad [70]	1 ResNet-50	**1.6GB**	66.4	59.2	**24.1**	34.2	26.8	47.6	22.1	**35.7**	31.7	37.3
Fork-Norm. MTL	1 ResNet-50	**1.6GB**	**67.3**	**59.9**	24.0	34.9	**28.9**	**48.3**	**22.3**	34.6	**33.3**	**38.8**

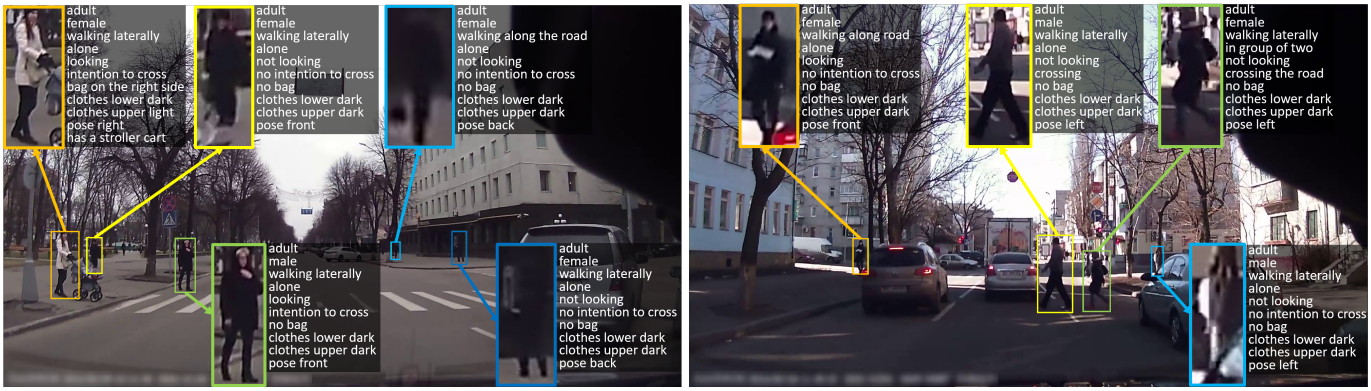


Fig. 4: **Qualitative examples.** Results of the proposed pedestrian detection with multiple attributes on two scenes from JAAD val set. All pedestrians of sufficient apparent size are correctly detected as indicated by colored bounding boxes (5 in the left scene, 4 in the right scene). For each detected pedestrian, an inset zooms in the image region and indicates the detected attributes among: 'age', 'gender', 'walking' and 'motion direction', 'group size', 'looking', 'road crossing intention' or 'instant road crossing', 'bag' (all types), 'upper clothing', 'lower clothing', 'back/front/left/right pose'. Difficult instances may not have ground truth annotation for some of the attributes, but the model still outputs plausible predictions.

separately. Note that the results seem stable when changing the batch size, and that the standard gradient accumulation variant with all tasks ($A = 32$) uses a learning rate halved to avoid unstable training due to exploding gradients. Task weights are learned using uncertainty [25] with bigger values for detection tasks to bias toward them as they are the most important ones (factors of 2 for $A = 3$ and $A = 4$, 4 for $A = 13$ and 7 for $A = 32$). The vote step uses threshold $\gamma = 0.2$, and OPTICS [3] with a minimum cluster size of 10, a maximum radius of 5, and cluster threshold of 0.5.

During training, images are augmented with random horizontal flipping, scaling to width 961px, cropping out of top third (making them 369px high), zoom in or out with a factor in [0.95, 1.05], random color jittering, random jpeg compression and random grayscale conversion. This yields final feature maps of size 121×47 neurons with our network.

c) Detection and Classification Metrics: We evaluate pedestrian detection with average precision (AP) at an Intersection-over-Union (IoU) threshold of 0.5. Each binary or categorical attribute is evaluated by computing similar APs considering all values of the attribute as different classes of pedestrians and taking the mean over them. For attribute *time-to-crossing*, we consider APs for 10 classes corresponding to an absolute error lower than 10 thresholds, varying from 0.5s to 5s by step of 0.5s, and average them again. A global mean AP (mAP) summarizing the performances across all attributes (depending on the set of tasks learned) is given by averaging detection AP and all attribute APs.

In order to enable comparison with box classification approaches (i.e., using ground-truth bounding boxes as input), we adapt our evaluation pipeline to fit this *classification* setup. Each ground-truth pedestrian is matched with the closest detection whose center is inside the bounding box, and the attribute predictions are taken from the matched detection. When no match is found, predictions are the classes with the most examples from the training set. Image-wise predictions for crossing forecasting are obtained from the detection with highest confidence on this attribute. From our understanding, competing methods either use ground truth boxes as input or evaluate only on the set of correctly detected instances, but the results are then heavily dependent on the detection performance. For the attributes *looking* and *walking*, the evaluation protocol of Rasouli et al. [47] is rather different, so we adopt a similar one. For each of these two attributes, results are computed on a balanced test set, where all ground truths of the class with fewer examples are kept, and the same number of ground truths are randomly sampled from the other class. Results are computed on predictions from the selected ground truths only. The sampling is done 10 times and APs are averaged. We denote these metrics with a star.

B. Motivation for Multi-Task Learning

In an autonomous vehicle, on-board hardware sets strict limits on the memory, the number of operations, the power consumption and the inference time for the models. A fair comparison should therefore take these constraints into account

TABLE II: **Evaluation of detection, box and image classification** on JAAD test set in accuracy (%) and average precision (%). Attributes learned by models are indicated in column 'Attributes', a ✓ in column 'Box' indicates that the method uses ground-truth bounding boxes as input, and a ✓ in column 'Video' indicates that the method is trained and evaluated on video sequences. Metrics with a star are evaluated on balanced sets.

	Model				Detection	Crossing				Looking	Walking
	Name	Attributes	Box	Video	Ped. AP	Box Acc.	Box AP	Img. Acc.	Img. AP	Box AP*	Box AP*
(a)	RetinaNet [44]	Det.			56.1	–	–	–	–	–	–
(b)	Action+Context [47]	Cross.+Look.+Walk.	✓	✓	–	–	62.7	–	–	80.2	83.5
(c)	SKLT [17]	Det.+Cross.			–	81.0	–	–	–	–	–
(d)				✓	–	88.0	–	–	–	–	–
(e)	ST-DenseNet [56]	Det.+Cross.		✓	–	–	73.8	–	–	–	–
(f)		Cross.	✓	✓	–	–	84.8	–	–	–	–
(g)	Res-EnDec [22]	Cross.		✓	–	–	–	67.4	81.1	–	–
(h)	LookingAhead [10]	Cross.		✓	–	–	–	–	86.7	–	–
(i)	MTL Baseline	Det.+All Att. ($A = 32$)			70.0	86.5	82.3	78.3	92.9	75.6	88.1
(j)	Gradient Fork-Power Norm.	Det.+All Att. ($A = 32$)			70.4	87.4	83.8	80.6	94.3	76.8	88.8
(k)	Gradient Fork-Power Norm.	Det.+Cross.+Look.+Walk.			71.0	87.2	82.7	81.8	94.6	82.8	87.3
(l)	Gradient Fork-Power Norm.	Det.+Cross.			70.8	87.0	83.1	82.9	94.9	–	–
(m)	Gradient Fork-Power Norm.	Det.+Look.			71.6	–	–	–	–	82.0	–
(n)	Gradient Fork-Power Norm.	Det.+Walk.			71.1	–	–	–	–	–	85.9
(o)	Gradient Fork-Power Norm.	Det.			71.1	–	–	–	–	–	–

in addition to the performances. Although results may change when the number of attributes learned increases, this would have to be balanced with run-time resource utilization in practice. In order to motivate a MTL approach, we compare in Table I our MTL network to a collection of single-task networks, with one model learned for each attribute. In this section, learning is done on the training set and evaluation on the validation set.

Our MTL model takes 1.6GB of memory to predict all 32 attributes. This means that, on average, multiple single-task networks should take no more than 50MB each, in order to equate the memory footprint of both approaches, which would yield very small networks and probably poor results. For quantitative comparison, we trained the collection of models with ResNet-18 as backbone networks, i.e., the smallest network available that still allows the same experimental setup, in particular, which has PifPaf pre-trained weights. Although this approach uses smaller networks, it is important to note that it still has a memory footprint of 38.4GB, i.e., 24 times bigger than our MTL model. Aside from memory, it does on par with the MTL baseline. However, our fork-normalization approach outperforms the collection of networks, with both less memory and better attribute APs. The results thus validate the use of MTL in embedded applications with run-time requirements.

We also compare with PCGrad [70], another contemporary approach from the literature to stabilize training and prevent negative transfer between tasks. PCGrad computes gradients for all tasks separately, and modifies them when pairs of gradients have negative scalar products, so that all gradients point in more similar directions. It is noticeable that PCGrad requires a separate backward pass for each task, which does not scale well to numerous tasks in terms of training time. Differently from it, fork-normalization only happens when joining gradients at the fork and does not bring overhead. We implement it on top of our MTL baseline model. Although it also improves on the baseline and the collection of networks, it is outperformed by our fork-normalization approach.

Two examples of predictions from our model are displayed in Figure 4. Overall, we observe that detections are accurately localized, with no or few false positives. Although qualitative evaluation of attributes is harder due to low resolution of most pedestrians, the predictions look generally convincing.

C. Comparison with the State of the Art

Comparisons with approaches from the literature are presented in Table II for detection and classification on attributes *crossing* (i.e., whether the pedestrian will cross the road in front of the vehicle), *looking* (i.e., eye contact with the vehicle), and *walking* (i.e., standing or walking posture). To the best of our knowledge, there is no other work jointly addressing both detection and attribute recognition that we could compare to. In this section, learning is done on the union of the training and validation sets, and evaluation on the testing set.

Pedestrian detection results are given with Pedestrian AP in the column Detection of Table II. To the best of our knowledge, only RetinaNet [35] (a) has been used for pedestrian detection on JAAD [44], and it is outperformed by all our models by significant margins, with more than 25% of improvement, validating our approach for detection. The reason is that field-based methods are well suited for low resolution scenarios, such as for autonomous vehicles, thanks to the spatial aggregation post-processing naturally leveraging context [30]. On the other hand, RetinaNet relies on prior box classification, which is limited in the case of pedestrians of small apparent sizes.

Table II also presents the performances for the three attributes evaluated by previous works in the remaining columns. Our approach either outperforms or performs on par with state-of-the-art methods that use ground-truth detections and/or videos, while still learning multiple additional attributes. On attribute *crossing*, SKLT [17] (c-d) slightly outperforms our method (j) by 0.7% when using video. However, on single frames as we do, it falls behind, 7.3% worse when compared to our model. When ST-DenseNet [56] (e-f) uses predicted boxes, we outperform it by 13.6%, even without video information. Even their best result with ground-truth boxes is only 1.2% better than ours.

TABLE III: **Effect of gradient merging** on JAAD val set. APs (%) are shown for some attributes, mAP (%) is the average of APs for all attributes evaluated (different for each group). TtC stands for Time-to-Crossing.

	Detection	Intention		Behavior			Appearance			All
Gradient Merging	Pedestrian	Crossing	TtC	Looking	Walking	Front Pose	Age	Gender	Phone	mAP
Detection Only ($A = 1$)										
Accumulation	68.1	–	–	–	–	–	–	–	–	68.1
Mean Loss	67.5	–	–	–	–	–	–	–	–	67.5
Fork-Sample Norm.	66.8	–	–	–	–	–	–	–	–	66.8
Fork-Random Norm.	67.5	–	–	–	–	–	–	–	–	67.5
Fork-Average Norm.	67.3	–	–	–	–	–	–	–	–	67.3
Fork-Power Norm.	**68.2**	–	–	–	–	–	–	–	–	**68.2**
Detection + Intention Attributes ($A = 3$)										
Accumulation	67.1	58.6	**23.7**	–	–	–	–	–	–	49.8
Mean Loss	66.1	57.4	23.4	–	–	–	–	–	–	49.0
Fork-Sample Norm.	66.6	57.8	23.3	–	–	–	–	–	–	49.2
Fork-Random Norm.	66.9	59.8	23.4	–	–	–	–	–	–	50.0
Fork-Average Norm.	66.8	60.5	**23.7**	–	–	–	–	–	–	**50.3**
Fork-Power Norm.	**67.2**	**60.6**	22.9	–	–	–	–	–	–	50.2
Detection + Intention and Behavior Attributes ($A = 13$)										
Accumulation	65.0	58.0	22.0	33.2	**29.9**	45.3	–	–	–	39.2
Mean Loss	65.5	56.4	22.3	31.3	26.8	42.8	–	–	–	38.4
Fork-Sample Norm.	60.1	53.4	20.4	30.6	25.2	40.0	–	–	–	35.9
Fork-Random Norm.	65.3	59.0	23.5	31.7	29.0	46.5	–	–	–	40.1
Fork-Average Norm.	65.7	59.6	**23.7**	32.4	29.4	48.8	–	–	–	40.7
Fork-Power Norm.	**66.9**	**60.2**	23.6	**33.7**	**29.9**	**49.3**	–	–	–	**41.4**
Detection + Intention, Behavior and Appearance Attributes ($A = 32$)										
Accumulation	66.3	56.5	21.9	34.4	28.4	45.3	21.1	32.1	32.2	36.6
Mean Loss	66.3	59.6	22.6	32.7	27.6	**48.3**	19.8	30.4	31.9	35.9
Fork-Sample Norm.	54.3	49.3	16.9	28.0	24.2	38.7	13.6	24.4	26.9	30.5
Fork-Random Norm.	65.4	59.4	23.5	32.2	29.4	47.4	21.9	31.3	31.5	36.7
Fork-Average Norm.	65.5	58.8	22.9	32.6	28.1	46.3	17.0	27.6	31.7	35.5
Fork-Power Norm.	**67.3**	**59.9**	**24.0**	**34.9**	**28.9**	**48.3**	**22.3**	**34.6**	**33.3**	**38.8**

Fork-normalization also compares favorably to the state of the art (g-h) on image-wise metrics. Regarding attributes *looking* and *walking*, we outperform Action+Context [47] (b) when using a similar evaluation protocol on balanced sets. Overall, our gradient fork-power normalization (j) better generalizes to many tasks than standard accumulation (i). It is noticeable that our approach is generic regarding the choice of attributes, and can predict any kind of attributes, while competing models are designed and optimized for the specific attributes they predict.

D. Ablation Study

We carry out several ablation studies to analyse the impact of the gradient merging operation depending on the number of tasks learned. In this section, learning is done on the training set and evaluation on the validation set.

Table III summarizes results of all the gradient merging variants on the four sets of tasks. Note that the mAP score cannot be compared between different sets of attributes as the averages do not cover the same numbers of values. Overall, the fork-sample version clearly yields the worse results among all methods, possibly due to the fact that only one task is considered per example for each backward pass, therefore discarding a lot of supervision and introducing some noise. The mean loss approach also underperforms, although the gap is not as large. Among the four remaining methods, the gradient fork-normalization variants generally improve on the standard gradient accumulation, and the fork-power normalization consistently obtains the best global mAP results, i.e., over all attributes, or very close. Moreover, the gaps with the standard gradient accumulation get bigger the more tasks are learned, from 0.1% to 6.0%. Note that the learning rate for gradient accumulation trained on all tasks is halved because of exploding gradient issues.

Table IV investigates the AP gap from learning a single attribute to all of them. Our proposed approach consistently sees lesser drops or higher gains, and is less sensitive to the number of tasks thanks to more stable gradients. These results confirm that normalizing gradients at the fork helps scaling to numerous tasks.

VI. Conclusions

We introduced a Multi-Task Learning (MTL) approach for joint pedestrian detection and attribute recognition. It relies on a bottom-up field formalism, particularly suited to the low resolution context of autonomous vehicles. We experimented with detection of up to 32 pedestrian attributes simultaneously. Our final model detecting 32 attributes outperforms the state-of-the-art (RetinaNet [35]) single-task pedestrian detection by more than 25%. By increasing the number of attributes learned by the network, we highlighted an issue linked to gradient scale

TABLE IV: **Impact of number of tasks** on JAAD val set. APs (%) are given for some attributes when learned only with detection ($A = 2$) or along with all attributes ($A = 32$), and AP gaps when scaling to all attributes are indicated to show the evolutions of performances.

Gradient Merging	Detection and Single Attribute ($A = 2$)	Detection and All Attributes ($A = 32$)	AP Gap
Crossing			
Accumulation	58.1	56.5	-1.6 (-2.8%)
Fork-Power Norm.	58.1	59.9	**+1.8** (**+3.1%**)
Looking			
Accumulation	34.0	34.4	+0.4 (+1.2%)
Fork-Power Norm.	34.2	34.9	**+0.7** (**+2.0%**)
Front Pose			
Accumulation	47.9	45.3	-2.6 (-5.4%)
Fork-Power Norm.	45.5	48.3	**+2.8** (**+6.2%**)
Gender			
Accumulation	32.1	32.1	0.0 (0.0%)
Fork-Power Norm.	32.7	34.6	**+1.9** (**+5.8%**)
Time-to-Crossing			
Accumulation	23.6	21.9	-1.7 (-7.2%)
Fork-Power Norm.	24.1	24.0	**-0.1** (**-0.4%**)
Walking			
Accumulation	26.2	28.4	+2.2 (+8.4%)
Fork-Power Norm.	26.0	28.9	**+2.9** (**+11.2%**)
Age			
Accumulation	20.6	21.1	+0.5 (+2.4%)
Fork-Power Norm.	18.8	22.3	**+3.5** (**+18.6%**)
Phone			
Accumulation	34.4	32.2	-2.2 (-6.4%)
Fork-Power Norm.	32.6	33.3	**+0.7** (**+2.1%**)

in MTL with numerous tasks. We solved it by normalizing back-propagation at the fork in the architecture, leading to a more stable training and better generalization to the addition of tasks. Although we only show results for this model in the context of pedestrian analysis, we think this approach can have applications in more general MTL frameworks, as the reasoning about gradient norms should be generic.

Over the past years, academics and industry have joined forces to make autonomous vehicles a reality and save human lives. While many challenges remain, we believe that pedestrian safety must be one of the highest priorities. For many years, researchers focused on simply detecting them. In this work, we argue that we need to go beyond simple detection and detect as many attributes as possible. These attributes will help better anticipate pedestrian behaviors. While our work will impact the safety of autonomous vehicles, it can also be used in any Advanced Driver Assistance Systems (ADAS). Anticipating early enough whether a pedestrian will cross the road will potentially save many lives. We hope that our work will foster more research in this area.

REFERENCES

[1] A. Alahi, V. Ramanathan, K. Goel, A. Robicquet, A. A. Sadeghian, L. Fei-Fei, and S. Savarese, "Learning to predict human behavior in crowded scenes," in *Group and Crowd Behavior for Computer Vision*. Elsevier, 2017, pp. 183–207. 1

[2] W. Alvarez, F. Moreno, O. Sipele, N. Smirnov, and C. Olaverri-Monreal, "Autonomous driving: Framework for pedestrian intention estimation in a real world scenario," in *IEEE Intelligent Vehicles Symposium (IV)*, 2020. 1

[3] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "OPTICS: Ordering points to identify the clustering structure," *ACM Sigmod Record (SIGMOD)*, vol. 28, no. 2, pp. 49–60, 1999. 4, 7

[4] E. Arnold, O. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, and A. Mouzakitis, "A survey on 3D objectddetection methods for autonomous driving applications," *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, vol. 20, no. 10, pp. 3782–3795, 2019. 1

[5] S. Bell, Y. Liu, S. Alsheikh, Y. Tang, E. Pizzi, M. Henning, K. Singh, O. Parkhi, and F. Borisyuk, "GrokNet: Unified computer vision model trunk and embeddings for commerce," in *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD)*, 2020. 2

[6] S. Bonnin, T. H. Weisswange, F. Kummert, and J. Schmüdderich, "Pedestrian crossing prediction using multiple context-based models," in *Proceedings of the International Conference on Intelligent Transportation Systems (ITSC)*, 2014, pp. 378–385. 2

[7] S. A. Bouhsain, S. Saadatnejad, and A. Alahi, "Pedestrian intention prediction: A multi-task perspective," *European Association for Research in Transportation conference (HEART)*, 2020. 1

[8] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proceedings of the IEEE European Conference on Computer Vision (ECCV)*, 2020. 3

[9] R. Caruana, "Multitask learning," *Machine Learning*, vol. 28, no. 1, pp. 41–75, 1997. 1, 2

[10] M. Chaabane, A. Trabelsi, N. Blanchard, and R. Beveridge, "Looking ahead: Anticipating pedestrians crossing with future frames prediction," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019. 1, 8

[11] F. Chabot, M. Chaouch, J. Rabarisoa, C. Teuliere, and T. Chateau, "Deep MANTA: A coarse-to-fine many-task network for joint 2D and 3D vehicle analysis from monocular image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2040–2049. 2

[12] H. Chen, A. Gallagher, and B. Girod, "Describing clothing by semantic attributes," in *Proceedings of the IEEE European Conference on Computer Vision (ECCV)*, 2012, pp. 609–623. 1

[13] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2018. 2, 5

[14] L. Claussmann, M. Revilloud, D. Gruyer, and S. Glaser, "A review of motion planning for highway autonomous driving," *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, 2019. 1

[15] P. Coscia, F. Castaldo, F. A. Palmieri, L. Ballan, A. Alahi, and S. Savarese, "Point-based path prediction from polar histograms," in *Proceedings of the International Conference on Information Fusion (FUSION)*. IEEE, 2016, pp. 1961–1967. 1

[16] Y. Deng, P. Luo, C. C. Loy, and X. Tang, "Pedestrian attribute recognition at far distance," in *Proceedings of ACM International Conference on Multimedia (MM)*, 2014, pp. 789–792. 1

[17] Z. Fang and A. López, "Is the pedestrian going to cross? answering by 2D pose estimation," in *IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 1271–1276. 2, 8

[18] Z. Fang, D. Vázquez, and A. López, "On-board detection of pedestrian intentions," *Sensors*, vol. 17, no. 10, p. 2193, 2017. 2

[19] L. Ferranti, B. Brito, E. Pool, Y. Zheng, R. Ensing, R. Happee, B. Shyrokau, J. Kooij, J. Alonso-Mora, and D. Gavrila, "SafeVRU: A research platform for the interaction of self-driving vehicles with vulnerable road users," in *IEEE Intelligent Vehicles Symposium (IV)*, 2019, pp. 1660–1666. 1

[20] G. Gkioxari, R. Girshick, and J. Malik, "Actions and attributes from wholes and parts," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 2470–2478. 1

[21] G. Gkioxari, R. Girshick, and J. Malik, "Contextual action recognition with R*CNN," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1080–1088. 1

[22] P. Gujjar and R. Vaughan, "Classifying pedestrian actions in advance using predicted video of urban driving scenes," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 2097–2103. 1, 8

[23] M. Guo, A. Haque, D.-A. Huang, S. Yeung, and L. Fei-Fei, "Dynamic task prioritization for multitask learning," in *Proceedings of the IEEE European Conference on Computer Vision (ECCV)*, 2018, pp. 270–287. 2, 5

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778. 6

[25] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 7482–7491. 2, 3, 4, 5, 7

[26] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015. 6

[27] S. Köhler, M. Goldhammer, S. Bauer, K. Doll, U. Brunsmann, and K. Dietmayer, "Early detection of the pedestrian's intention to cross the street," in *Proceedings of the International Conference on Intelligent Transportation Systems (ITSC)*, 2012, pp. 1759–1764. 2

[28] I. Kokkinos, "UberNet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2, 4, 5

[29] P. Kothari, S. Kreiss, and A. Alahi, "Human trajectory forecasting in crowds: A deep learning perspective," *arXiv preprint arXiv:2007.03639*, 2020. 1

[30] S. Kreiss, L. Bertoni, and A. Alahi, "PifPaf: Composite fields for human pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 11 977–11 986. 1, 2, 3, 6, 8

[31] S. Kuutti, R. Bowden, Y. Jin, P. Barber, and S. Fallah, "A survey of deep learning applications to autonomous vehicle control," *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, 2019. 1

[32] R. Layne, T. Hospedales, and S. Gong, "Person re-identification by attributes," in *Proceedings of the British Machine Vision Conference (BMVC)*, 2012. 2

[33] D. Li, Z. Zhang, X. Chen, and K. Huang, "A richly annotated pedestrian dataset for person retrieval in real surveillance scenarios," *IEEE Transactions on Image Processing (TIP)*, vol. 28, no. 4, pp. 1575–1590, 2019. 1

[34] Y. Li, C. Huang, C. C. Loy, and X. Tang, "Human attribute recognition by deep hierarchical contexts," in *Proceedings of the IEEE European Conference on Computer Vision (ECCV)*, 2016. 1

[35] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988. 6, 8, 9

[36] Y. Lin, L. Zheng, Z. Zheng, Y. Wu, Z. Hu, C. Yan, and Y. Yang, "Improving person re-identification by attribute and identity learning," *Pattern Recognition (PR)*, vol. 95, pp. 151–161, 2019. 2

[37] B. Liu, E. Adeli, Z. Cao, K.-H. Lee, A. Shenoi, A. Gaidon, and J. C. Niebles, "Spatiotemporal relationship reasoning for pedestrian intent prediction," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2020. 1

[38] S. Liu, E. Johns, and A. Davison, "End-to-end multi-task learning with attention," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1871–1880. 2

[39] Y. Lu, A. Kumar, S. Zhai, Y. Cheng, T. Javidi, and R. Feris, "Fully-adaptive feature sharing in multi-task networks with applications in person attribute classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1131–1140. 2

[40] S. Malla, B. Dariush, and C. Choi, "TITAN: Future forecast using action priors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11 186–11 196. 1

[41] C. Mao, A. Gupta, V. Nitin, B. Ray, S. Song, J. Yang, and C. Vondrick, "Multitask learning strengthens adversarial robustness," in *Proceedings of the IEEE European Conference on Computer Vision (ECCV)*, 2020. 2

[42] E. Meyerson and R. Miikkulainen, "Beyond shared hierarchies: Deep multitask learning through soft layer ordering," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018. 2

[43] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert, "Cross-stitch networks for multi-task learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3994–4003. 2

[44] D. O. Pop, A. Rogozan, C. Chatelain, F. Nashashibi, and A. Bensrhair, "Multi-task deep learning for pedestrian detection, action recognition and time to cross prediction," *IEEE Access*, vol. 7, pp. 149 318–149 327, 2019. 2, 8

[45] R. Quintero, I. Parra, D. Fernández-Llorca, and M. Sotelo, "Pedestrian intention and pose prediction through dynamical models and behaviour classification," in *Proceedings of the International Conference on Intelligent Transportation Systems (ITSC)*, 2015, pp. 83–88. 2

[46] A. Rasouli, I. Kotseruba, T. Kunic, and J. Tsotsos, "PIE: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 6262–6271. 1

[47] A. Rasouli, I. Kotseruba, and J. Tsotsos, "Are they going to cross? a benchmark dataset and baseline for pedestrian crosswalk behavior," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 206–213. 1, 2, 6, 7, 8, 9

[48] A. Rasouli, I. Kotseruba, and J. Tsotsos, "Pedestrian action anticipation using contextual feature fusion in stacked RNNs," in *Proceedings of the British Machine Vision Conference (BMVC)*, 2019. 1

[49] A. Rasouli, I. Kotseruba, and J. K. Tsotsos, "It's not all about size: On the role of data properties in pedestrian detection," in *Proceedings of the IEEE European Conference on Computer Vision (ECCV)*, 2018. 2

[50] A. Rasouli and J. Tsotsos, "Autonomous vehicles that interact with pedestrians: A survey of theory and practice," *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, 2019. 1

[51] E. Rehder and H. Kloeden, "Goal-directed pedestrian prediction," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 50–58. 1

[52] E. Rehder, F. Wirth, M. Lauer, and C. Stiller, "Pedestrian prediction by planning using deep neural networks," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 1–5. 1

[53] D. Ridel, N. Deo, D. Wolf, and M. Trivedi, "Understanding pedestrian-vehicle interactions with vehicle mounted vision: An LSTM model and empirical analysis," in *IEEE Intelligent Vehicles Symposium (IV)*, 2019. 1

[54] D. Ridel, E. Rehder, M. Lauer, C. Stiller, and D. Wolf, "A literature review on the prediction of pedestrian behavior in urban scenarios," in *Proceedings of the International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 3105–3112. 1

[55] K. Saleh, M. Hossny, and S. Nahavandi, "Intent prediction of vulnerable road users from motion trajectories using stacked LSTM network," in *Proceedings of the International Conference on Intelligent Transportation Systems (ITSC)*, 2017, pp. 327–332. 1

[56] K. Saleh, M. Hossny, and S. Nahavandi, "Real-time intent prediction of pedestrians for autonomous ground vehicles via spatio-temporal DenseNet," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2019. 1, 8

[57] F. Schneemann and P. Heinemann, "Context-based detection of pedestrian crossing intention for autonomous driving in urban environments," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 2243–2248. 2

[58] A. Schumann and R. Stiefelhagen, "Person re-identification by deep learning attribute-complementary information," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 20–28. 2

[59] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1874–1883. 6

[60] M. D. Sulistiyo, Y. Kawanishi, D. Deguchi, I. Ide, T. Hirayama, J.-Y. Zheng, and H. Murase, "Attribute-aware loss function for accurate semantic segmentation considering the pedestrian orientations," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 103, no. 1, pp. 231–242, 2020. 1, 2

[61] M. Suteu and Y. Guo, "Regularizing deep multi-task networks using orthogonal gradients," *arXiv preprint arXiv:1912.06844*, 2019. 2

[62] C. Tang, L. Sheng, Z. Zhang, and X. Hu, "Improving pedestrian attribute recognition with weakly-supervised multi-scale attribute-specific

localization," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 4997–5006. 1

[63] Y. Tian, P. Luo, X. Wang, and X. Tang, "Pedestrian detection aided by deep learning semantic tasks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 5079–5087. 1, 2

[64] S. Vandenhende, S. Georgoulis, and L. Van Gool, "MTI-Net: Multi-scale task interaction networks for multi-task learning," in *Proceedings of the IEEE European Conference on Computer Vision (ECCV)*, 2020. 2

[65] D. Varytimidis, F. Alonso-Fernandez, B. Duran, and C. Englund, "Action and intention recognition of pedestrians in urban traffic," in *Proceedings of the International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, 2018, pp. 676–682. 1, 2

[66] B. Völz, K. Behrendt, H. Mielenz, I. Gilitschenski, R. Siegwart, and J. Nieto, "A data-driven approach for pedestrian intention estimation," in *Proceedings of the International Conference on Intelligent Transportation Systems (ITSC)*, 2016, pp. 2607–2612. 1

[67] B. Völz, H. Mielenz, G. Agamennoni, and R. Siegwart, "Feature relevance estimation for learning pedestrian behavior at crosswalks," in *Proceedings of the International Conference on Intelligent Transportation Systems (ITSC)*, 2015, pp. 854–860. 1, 2

[68] Z. Wang and N. Papanikolopoulos, "Estimating pedestrian crossing states based on single 2D body pose," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2020. 2

[69] Y. Yang and T. Hospedales, "Deep multi-task representation learning: A tensor factorisation approach," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017. 2

[70] T. Yu, S. Kumar, A. Gupta, S. Levine, K. Hausman, and C. Finn, "Gradient surgery for multi-task learning," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 2, 7, 8

[71] A. Zamir, A. Sax, W. Shen, L. Guibas, J. Malik, and S. Savarese, "Taskonomy: Disentangling task transfer learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2

[72] J. Zhang, L. Lin, Y. Li, Y.-c. Chen, J. Zhu, Y. Hu, and S. Hoi, "Attribute-aware pedestrian detection in a crowd," *arXiv preprint arXiv:1910.09188*, 2019. 1, 2

[73] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," *arXiv preprint arXiv:1904.07850*, 2019. 3

Taylor Mordan received the engineering degree from ENSTA ParisTech, Paris, France, and the M.S. degree in computer science from UPMC, Paris, France, in 2015, then the Ph.D. degree in computer science from Sorbonne University, Paris, France, in 2018. From 2015 to 2018, he was a Research Assistant with Thales LAS France. Since 2019, he has been a Post-Doctoral Researcher with VITA lab, EPFL, Lausanne, Switzerland. His research interests include computer vision, multi-task learning, and applications to perception in autonomous vehicles.

Matthieu Cord is full professor at Sorbonne University. He is also part-time principal scientist at Valeo.ai. His research expertise includes computer vision, machine learning and artificial intelligence. He is the author of more 150 publications on image classification, segmentation, deep learning, and multimodal vision and language understanding. He is an honorary member of the Institut Universitaire de France and served from 2015 to 2018 as an AI expert at CNRS and ANR (National Research Agency).

Patrick Pérez is Scientific Director of Valeo.ai, a Valeo research lab on artificial intelligence for automotive applications. Before joining Valeo, Patrick Pérez has been Distinguished Scientist at Technicolor (2009-2918), researcher at Inria (1993-2000, 2004-2009) and at Microsoft Research Cambridge (2000-2004). His research revolves around machine learning for scene understanding, data mining and visual editing.

Alexandre Alahi is currently an Assistant Professor at EPFL. He spent five years at Stanford University as a Post-doc and Research Scientist after obtaining his Ph.D. from EPFL. His research enables machines to perceive the world and make decisions in the context of transportation problems and smart environments. He has worked on the theoretical challenges and practical applications of socially-aware Artificial Intelligence, i.e., systems equipped with perception and social intelligence.