Cutting Edge: Soft Correspondences in Multimodal Scene Parsing

Sarah Taghavi Namin[1,2] Mohammad Najafi[1,2] Mathieu Salzmann[2,3] Lars Petersson[1,2]
[1]Australian National University (ANU) [2]NICTA* [3]CVLab, EPFL, Switzerland
{sarah.namin, mohammad.najafi, lars.petersson}@nicta.com.au mathieu.salzmann@epfl.ch

Abstract

Exploiting multiple modalities for semantic scene parsing has been shown to improve accuracy over the single-modality scenario. Existing methods, however, assume that corresponding regions in two modalities have the same label. In this paper, we address the problem of data misalignment and label inconsistencies, e.g., due to moving objects, in semantic labeling, which violate the assumption of existing techniques. To this end, we formulate multimodal semantic labeling as inference in a CRF, and introduce latent nodes to explicitly model inconsistencies between two domains. These latent nodes allow us not only to leverage information from both domains to improve their labeling, but also to cut the edges between inconsistent regions. To eliminate the need for hand tuning the parameters of our model, we propose to learn intra-domain and inter-domain potential functions from training data. We demonstrate the benefits of our approach on two publicly available datasets containing 2D imagery and 3D point clouds. Thanks to our latent nodes and our learning strategy, our method outperforms the state-of-the-art in both cases.

1. Introduction

Multi-modal scene analysis aims at leveraging complementary information captured by multiple sensing modalities, such as 3D LIDAR and 2D imagery. In the context of semantic labeling, where the goal is to assign a class label to the elements of each modality, such as image pixels and 3D points, this has been shown to consistently yield increased accuracy over relying on a single domain [7, 25, 5, 2, 15, 17].

Nevertheless, existing methods suffer from an important limitation: they typically assume that corresponding regions in two modalities always have the same label. This assumption is encoded either explicitly by having a single label variable for all modalities [7, 5, 2], or implicitly by penalizing label differences between the domains [25, 15, 17].

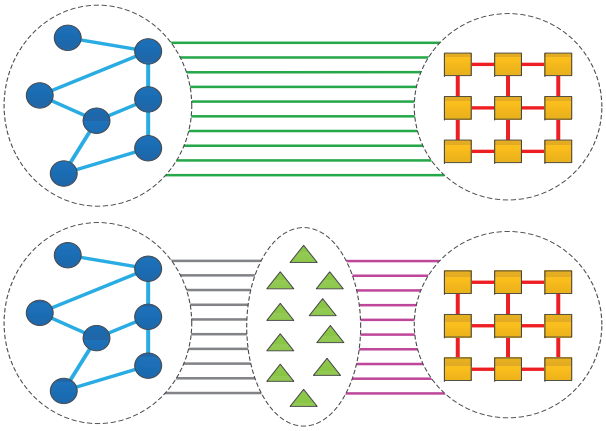

Figure 1. **Top:** Existing approaches typically directly connect corresponding regions in different modalities and penalize these regions for taking different labels, thus producing wrong labeling in the presence of data misalignment, or other causes of label disagreement. **Bottom:** Here, we introduce latent nodes that are placed between each connected pair of 2D and 3D nodes in the graph and explicitly let us account for such inconsistencies, and potentially cut edges between the different domains. Blue spheres denote one domain (e.g., 3D) and orange squares another one (e.g., 2D). Our latent nodes are represented as green triangles.

While this assumption may seem reasonable, it is often violated in realistic scenarios. Indeed, in practice, the different modalities are typically not perfectly aligned/registered. Furthermore, in dynamic scenes, moving objects may not easily be captured by some sensors, such as 3D LIDAR, due to their lower acquisition speed. To give a concrete example, in the NICTA/2D3D dataset employed in our experiments, 17% of the connections between the two modalities correspond to inconsistent labels. As a consequence, since they fail to model these inconsistencies, existing methods will typically produce wrong labels in at least one modality.

In this paper, we introduce an approach to multimodal semantic labeling that explicitly accounts for the inconsistencies of the domains. To this end, as illustrated in Fig. 1, we formulate multimodal scene parsing as inference in a Conditional Random Field (CRF), and introduce latent nodes to handle conflicting evidence between the different domains. The benefit of these latent nodes is twofold: First,

*NICTA is funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program.

† The authors would like to thank Justin Domke for his assistance in implementing the learned potentials.

they can leverage information from both domains to improve their respective labeling. Second, and maybe more importantly, these nodes allow us to cut the edges between regions in different modalities when the local evidence of the domains is inconsistent. As a result, our approach lets us correctly assign different labels to the modalities.

More specifically, each connection between two domains is encoded by a latent node, which can take either a label from the same set as the regular nodes, or an additional label that explicitly represents a broken link. We then model the connections between the latent nodes and the different modalities with potential functions that allow us to handle inconsistencies. While many such connections exist, they come at little cost, because the only cases of interest are when the latent node and the regular node have the same label, and when the latent node indicates a broken edge. By contrast, having direct links between two modalities would require to consider potential functions for each combination of two labels (i.e., for L labels, L^2 vs $2L$ in our model). Furthermore, we also model intra-domain connections with potential functions that encode some notion of label compatibility and thus let us model more accurately the relationships between different class labels. Altogether, these connections allow the information to be transferred across the domains, thus encoding the fact that some classes may be easier to recognize in one modality than in the others. Since such general potential functions cannot realistically be manually tuned, we propose to learn them from training data. To this end, we make use of the truncated tree-reweighted (TRW) learning algorithm of [4]. The resulting method therefore incorporates local evidence from each domain, intra-domain relationships and inter-domain compatibility via our latent nodes.

We demonstrate the effectiveness of our approach on two publicly available 2D-3D scene analysis datasets: The NICTA/2D3D dataset [17] and the CMU/VMR dataset [15]. Our experiments evidence the benefits of our latent nodes and of learning potentials for multimodal scene parsing. In particular, our approach outperforms the state-of-the-art on both datasets.

2. Related Work

Semantic scene analysis has been an important problem in computer vision for the past decade. In particular, scene parsing from 2D imagery has been intensely studied, yielding increasingly accurate results [20, 23, 11, 6, 24, 8]. With the advent of 3D depth sensors, such as laser range sensors (LIDAR) and RGB-D cameras (e.g., Kinect), it seems natural to try and leverage these additional sources of information to reach even better levels of scene understanding [16, 1, 19, 13].

As a matter of fact, combining 2D imagery and 3D point clouds for semantic labeling has been the focus of several recent works [7, 5, 25, 2, 15, 17]. In particular, [7, 5] defined models on the variables corresponding to only one modality and augmented them with information extracted from the other domain. This approach, however, assumes that the same portions of the scene are observed in both modalities, which is virtually never the case in practice. By contrast, the model of [2] incorporates variables for the two domains, but still relies on a single variable for the corresponding regions in both modalities. Therefore, this model still assumes that the modalities are perfectly aligned. This, unfortunately, can typically not be achieved in practice, and the above-mentioned techniques will thus misclassify some regions in at least one of the domains.

Some approaches have nonetheless proposed to relax this assumption by having separate variables for the two modalities, even for corresponding regions. In this context, [15] designed a hierarchical labeling approach that alternatively performs classification in each domain. However, since the classification result of one modality is then transferred to help labeling in the other domain (depending on the overlap area of the projection of the 3D segment onto the 2D region), this method implicitly encodes the assumption that corresponding regions should have the same label. In [25], a framework to train a joint 2D-3D graph from unlabeled data was proposed. As in [15], this framework transfers the labels from one modality to the other, thus implicitly assuming that corresponding 2D and 3D nodes belong to the same class. Similarly, in [17], a multimodal graphical model with separate nodes for each modality was introduced. This method, however, relied on a Potts model as pairwise potentials for both intra-domain and inter-domain edges. As a consequence, it also implicitly attempts to assign the same label to the corresponding nodes in each modality.

Here, by contrast, we propose to introduce latent nodes in a CRF to explicitly model the inconsistencies between two modalities. Furthermore, our approach lets us learn the intra-domain and inter-domain relationships from training data. Learning the parameters of CRFs for semantic labeling has been tackled by a number of works, such as [22, 10] with mean-field inference, [12] with TRW, and [18] for loopy belief propagation. Of more specific interest to us is the problem of learning label compatibility, as studied by [10] for 2D images and by [9] for 3D data. Here, we consider label compatibility within and across domains. To the best of our knowledge, this is the first time such a learning approach is employed for multimodal scene parsing.

3. A Multimodal CRF with Latent Nodes

We now introduce our approach to multimodal semantic labeling in the presence of inconsistencies across the domains. We focus the discussion on two modalities, 2D imagery and 3D point clouds, which are typically the most popular ones for scene parsing. Note, however, that our

approach generalizes to other modalities, such as hyperspectral or infrared data.

As mentioned above, we formulate multimodal semantic labeling as inference in a CRF. Our CRF contains separate nodes for 2D regions (i.e., superpixels) and 3D regions (i.e., 3D segments). More details about these regions are provided in Section 5. In addition to these nodes, we propose to introduce latent nodes that allow us to account for inconsistencies between the different domains. To this end, and as illustrated in Fig. 1, we incorporate one such latent node between each pair of corresponding 2D and 3D nodes. This results in edges between either a 2D node and a latent node, or a 3D node and a latent node, but no edges directly connecting a 2D node to a 3D node. Our latent nodes can then either take a label from the same space as the 2D and 3D nodes, or take another label indicating that the link between the two modalities should be cut. Figs. 2 and 3 illustrate how our latent nodes operate in the case of data misalignment and moving objects, respectively.

Formally, let $\mathbf{y}^{2D} = \{y_{ij}^{2D}\}$, $1 \leq i \leq F$, $1 \leq j \leq N_i$ be the set of variables encoding the labels of the 2D nodes in F frames, with frame i containing N_i 2D regions. Similarly, let $\mathbf{y}^{3D} = \{y_i^{3D}\}$, $1 \leq i \leq M$ be the set of variables encoding the label of M 3D nodes. Each of these variables, either 2D or 3D, can take a label in the set $\mathcal{L} = \{1, \cdots, L\}$. Furthermore, let T be the number of pairs of corresponding 2D and 3D nodes, found in the manner described in Section 5. We then denote by $\mathbf{y}^{\Delta} = \{y_t^{\Delta}\}$, $1 \leq t \leq T$ the latent nodes associated with these correspondences. These variables can be assigned a label from the space $\mathcal{L}' = \{0, 1, \cdots, L\}$.

Given features extracted from the 2D and 3D regions, $\mathbf{x}^{2D} = \{\mathbf{x}_{ij}^{2D}\}$ and $\mathbf{x}^{3D} = \{\mathbf{x}_i^{3D}\}$, respectively, the joint distribution of the 2D, 3D and latent nodes conditioned on the features can be expressed as

$$P(\mathbf{y}^{2D}, \mathbf{y}^{3D}, \mathbf{y}^{\Delta} | \mathbf{x}^{2D}, \mathbf{x}^{3D}) = \frac{1}{Z} \cdot \tag{1}$$

$$\exp\Big(-\sum_{i=1}^{F}\sum_{j=1}^{N_i}\Phi_{ij}^{2D} - \sum_{i=1}^{M}\Phi_i^{3D} - \sum_{t=1}^{T}\Phi_t^{\Delta} - \sum_{i=1}^{F}\sum_{(j,k)\in\mathcal{E}_i^{2D}}\Psi_{ijk}^{2D}$$

$$-\sum_{(i,j)\in\mathcal{E}^{3D}}\Psi_{ij}^{3D} - \sum_{i=1}^{F}\sum_{(j,t)\in\mathcal{E}^{2D-\Delta}}\Psi_{ijt}^{2D-\Delta} - \sum_{(i,t)\in\mathcal{E}^{3D-\Delta}}\Psi_{it}^{3D-\Delta}\Big),$$

where Z is the partition function, and where Φ^{2D}, Φ^{3D}, and Φ^{Δ} denote the unary potentials of the 2D, 3D and latent nodes, respectively. Ψ^{2D}, Ψ^{3D}, $\Psi^{2D-\Delta}$ and $\Psi^{3D-\Delta}$ denote pairwise potentials defined over the set of edges $\mathcal{E}^{2D}$, $\mathcal{E}^{3D}$, $\mathcal{E}^{2D-\Delta}$ and $\mathcal{E}^{3D-\Delta}$, respectively. To obtain a labeling, we perform inference in our CRF by making use of the truncated TRW algorithm of [4]. In the remainder of this section, we discuss the different potentials of Eq. 1 in details.

Unary potentials:
The unary potential of a node indicates the cost of assigning each label to the node, and is typically computed from local evidence. For the 2D and 3D nodes, we define the cost of assigning label l to the corresponding variables as

$$\Phi_{ij}^{2D}(y_{ij}^{2D} = l) = \mathbf{A}_l^{2D}\mathbf{x}_{ij}^{2D} , \tag{2}$$

and

$$\Phi_i^{3D}(y_i^{3D} = l) = \mathbf{A}_l^{3D}\mathbf{x}_j^{3D} , \tag{3}$$

respectively, where $\mathbf{A}^{2D} \in \mathbb{R}^{L\times D_{2D}}$ and $\mathbf{A}^{3D} \in \mathbb{R}^{L\times D_{3D}}$ are the parameter matrices for the 2D and 3D unary potentials, with $\mathbf{A}_l^{2D}$ the row of $\mathbf{A}^{2D}$ corresponding to label l. Since they directly act on the local features $\mathbf{x}_{ij}^{2D}$ and $\mathbf{x}_i^{3D}$, these matrices encode how much each feature dimension should be relied on to predict a specific label. Note that D_{2D} and D_{3D} refer to the dimensions of the 2D and 3D feature vectors, respectively.

Similarly, the unary potential for the latent nodes is defined as

$$\Phi_t^{\Delta}(y_t^{\Delta} = l) = \mathbf{A}_l^{\Delta}\mathbf{x}_t^{\Delta} , \tag{4}$$

where $\mathbf{A}^{\Delta}$ is again a parameter matrix, which this time contains $L + 1$ rows to represent the fact that a latent node can take on an additional label to cut the connection between a 2D and a 3D node. The feature vector of a latent node is constructed by concatenating the features of the corresponding 2D and 3D nodes, i.e., $\mathbf{x}_t^{\Delta} = [(\mathbf{x}_{ij}^{2D})^T, (\mathbf{x}_k^{3D})^T]^T$. Having access to both 2D and 3D features allows this unary to detect mismatches in the 2D and 3D observations, and, in that event, favor cutting the corresponding edge.

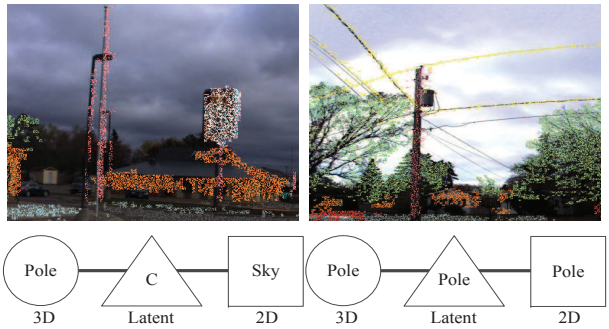

Figure 2. **Latent nodes for data misalignment. Left:** The projection of *pole* from 3D to 2D covers some regions of *sky*, which creates a connection between the corresponding 3D and 2D nodes. Having access to both 3D and 2D features, the latent node should detect the mismatch and cut this connection thus allowing the nodes to take different labels. **Right:** In this case, the projection is accurate. Therefore, the 2D and 3D features are both coherent with the class label *pole*, and thus the latent node should keep the edge active and predict the same class.

Pairwise potentials:
Pairwise potentials express the cost of all possible joint label assignments for two adjacent nodes in the graph. Here, in contrast with existing techniques [17] that rely on the

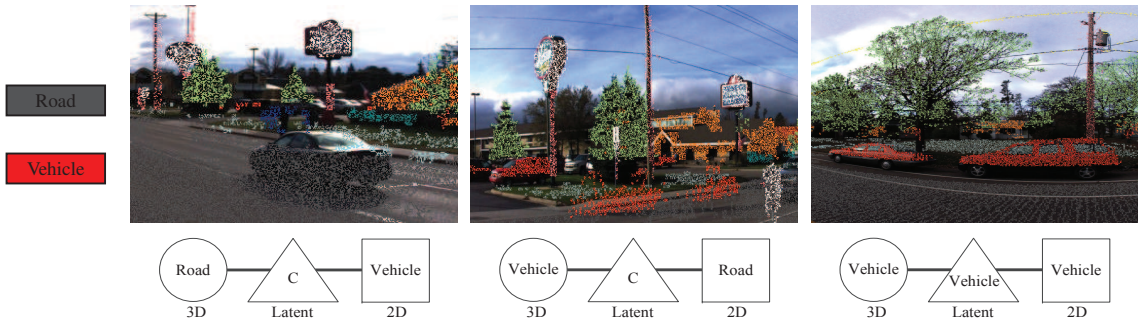

Figure 3. **Latent nodes for moving objects. Left:** A *vehicle* can be observed in 2D, but was not present when the 3D laser sensor covered this area. Therefore, the label of the 3D points is *road* instead of *vehicle* for 2D. By relying on both 2D and 3D features, the latent node should predict that this connection must be cut. **Middle:** This represents the opposite scenario where the image depicts an empty *road*, while the 3D points were acquired when a *vehicle* was passing. Here again, the latent node should cut the edge, thus allowing the nodes to take different labels. **Right:** In contrast, here, the 2D and 3D regions belong to the same class and thus have coherent features. The latent node should therefore leverage this information to help predicting the correct class *vehicle*.

simple Potts model and are thus limited to simply encouraging the nodes to share the same labels, we define general pairwise potentials that let us encode sophisticated label compatibilities. For the intra-domain edges, these potentials are defined as

$$\Psi^{2D}_{ijk}(y^{2D}_{ij} = l, y^{2D}_{ik} = m) = \mathbf{B}^{2D}_{lm}\mathbf{v}^{2D}_{ijk} \,, \tag{5}$$

and

$$\Psi^{3D}_{jk}(y^{3D}_{j} = l, y^{3D}_{k} = m) = \mathbf{B}^{3D}_{lm}\mathbf{v}^{3D}_{jk} \,, \tag{6}$$

where $\mathbf{B}^{2D}$ and $\mathbf{B}^{3D}$ are parameter matrices with L^2 rows representing all possible combinations of two labels, and $\mathbf{B}^{2D}_{lm}$ is the row of $\mathbf{B}^{2D}$ corresponding to the combination of label l with label m. In this case, as features $\mathbf{v}^{2D}_{ijk}$ and $\mathbf{v}^{3D}_{jk}$, we used the ℓ_2-norm of the difference of a subset of the original node features, which will be discussed in Section 5.

Similarly, the two pairwise potentials associated with the latent nodes that connect the 2D and 3D domains are defined as

$$\Psi^{2D-\Delta}_{ijt}(y^{2D}_{ij} = l, y^{\Delta}_{t} = m) = \mathbf{B}^{2D-\Delta}_{lm}\mathbf{v}^{2D-\Delta}_{ijt} \,, \tag{7}$$

and

$$\Psi^{3D-\Delta}_{jt}(y^{3D}_{j} = l, y^{\Delta}_{t} = m) = \mathbf{B}^{3D-\Delta}_{lm}\mathbf{v}^{3D-\Delta}_{jt} \,, \tag{8}$$

where the parameter matrices now have $L \times (L+1)$ rows to account for the extra label of the latent nodes, and where, in practice, we set the feature vectors $\mathbf{v}^{2D-\Delta}_{ijt}$ and $\mathbf{v}^{3D-\Delta}_{jt}$ to a single value of 1, thus resulting in $L \times (L+1)$ parameters. Note, however, that the effective number of parameters corresponding to these potentials is much smaller. The reason is that the only cases of interest are when the latent node and the regular node take the same label, and when the latent node indicates a broken link. The cost of the other label combinations should be heavily penalized since they never occur in practice. This therefore truly results in $2L$ parameters for each of these potentials.

4. Training our Multimodal Latent CRF

Our multimodal CRF contains many parameters, which thus cannot be tuned manually. Here, we propose to learn these parameters from training data. To this end, we make use of the direct loss minimization method of [4].

More specifically, let $\{\mathbf{z}_i\}$, $1 \leq i \leq N$ be a set of N labeled training examples, such that $\mathbf{z}_i = (\mathbf{x}^{2D}_i, \mathbf{x}^{3D}_i, \tilde{\mathbf{y}}^{2D}_i, \tilde{\mathbf{y}}^{3D}_i, \tilde{\mathbf{y}}^{\Delta}_i)$, where, with a slight abuse of notation compared to Section 3, $\mathbf{x}^{2D}_i$, resp. $\tilde{\mathbf{y}}^{2D}_i$, englobes the features, resp. ground-truth labels, of all the nodes in the i^{th} training sample, and similarly for the other terms in $\mathbf{z}_i$. In practice, to obtain the ground-truth labels of the latent nodes $\tilde{\mathbf{y}}^{\Delta}_i$, we simply check if the ground-truth labels of the corresponding 2D and 3D nodes agree, and set the label of the latent node to the same label if they do, and to 0 otherwise.

Learning the parameters of our model is then achieved by minimizing the empirical risk

$$r(\Theta) = \sum_{i=1}^{N} l(\Theta, \mathbf{z}_i) \tag{9}$$

w.r.t. $\Theta = \{\mathbf{A}^{2D}, \mathbf{A}^{3D}, \mathbf{A}^{\Delta}, \mathbf{B}^{2D}, \mathbf{B}^{3D}, \mathbf{B}^{2D-\Delta}, \mathbf{B}^{3D-\Delta}\}$, where $l(\Theta, \mathbf{z}_i)$ is a loss function.

Here, we use a marginal-based loss function, which measures how well the marginals obtained via inference in the model match the ground-truth labels. In particular, we rely on a loss function defined on the clique marginals [21]. This can be expressed as $l(\Theta, \mathbf{z}_i) = -\sum_c \log \mu(\mathbf{z}_{i,c}; \Theta)$ where c sums over all the cliques in the CRF, i.e., all the inter-domain and intra-domain pairwise cliques in our case, $\mathbf{z}_{i,c}$ denotes the variables of $\mathbf{z}_i$ involved in a particular clique c,

Note that our nodes are latent in the sense that they do not correspond to physical entities, not in the sense that we do not have access to their ground-truth during training.

and $\mu(\mathbf{z}_{i,c};\Theta)$ indicates the marginals of clique c obtained by performing inference with parameters Θ.

We use the publicly available implementation of [4] with truncated TRW as inference method. This method was shown to converge to stable parameters in only a few iterations. In practice, we run a maximum of 5 iterations of this algorithm.

5. Experiments

We evaluate our method on two publicly available 2D-3D multimodal datasets (NICTA/2D3D [17] and CMU/VMR [15]) and compare it to the state-of-the-art algorithms of [17] and [15]. In addition to these two baselines, we also provide the results of pairwise models with learned potentials acting on a single domain, either 2D or 3D. We will refer to these models as *Pairwise 2D (learned)* and *Pairwise 3D (learned)*. Furthermore, to evidence the effectiveness of our latent nodes, we also compare our approach to the same model as ours, but without latent nodes. This model therefore relies on learned pairwise potentials that directly connect corresponding 2D and 3D nodes (i.e., the same connections as in our model, but without going via the latent nodes). These potentials have a similar form as those in Eqs. 5, 6, 7 and 8, with a parameter matrix containing L^2 rows to encode all possible label combinations, and with features obtained by concatenating a subset of the 2D and 3D features (details below). In our results, we refer to this baseline as *No Latent*. Note that, while we treat this model as a baseline, it has never been published in the literature, and therefore can, in some sense, also be considered as a contribution of this paper. We followed the evaluation protocol of [17] and partitioned the data into 4 non-overlapping folds. We then used three of the folds for training and the remaining fold as test set. Below, we first provide some details regarding our features and potentials, and then discuss our results.

5.1. Features and Potentials

3D nodes:
We extracted the following 3D shape features from the point cloud data: Fast Point Feature Histogram descriptors, eigenvalue descriptors, deviation from the vertical axis and height. The 3D segments were obtained from these features by first classifying them using an SVM and then performing k-means clustering on the resulting classes. We then further leveraged the SVM results and used the negative logarithm of the multi-class SVM probabilities as features in our unary potentials. The probabilities for a segment were obtained by averaging over the points belonging to the segment. We also used three eigenvalue descriptors and the vertical-axis deviation as additional features.

2D nodes:
As 2D regions, we employed superpixels extracted by the mean-shift algorithm [3]. As in the 3D case, we utilized histogram of SIFT features [14], GLCM features and RGB values to train an SVM classifier, and used the negative logarithm of the SVM probabilities as features in our unary potentials. We augmented these features with six GLCM features and three RGB features.

Latent nodes:
The features of the latent nodes were obtained by concatenating the features of their respective 2D and 3D nodes, described above. Furthermore, we augmented these features with the normalized overlap area of the projection of the 3D segment onto the 2D superpixel.

Edges:
For the intra-domain potentials, we employed the ℓ_2-norm of the difference of a subset of the local feature vectors (RGB for 2D-2D edges and vertical-axis deviation for 3D-3D edges) as pairwise features. The feature vectors of the 2D-Δ and 3D-Δ edges was set to a single value of 1. In the case of the baseline model with no latent nodes, however, the feature vector of the 2D-3D edges was constructed by concatenating the RGB values of the 2D node with the eigenvalue features and the vertical-axis deviation of the 3D node, as well as with the same normalized overlap area used for the unary of the latent nodes. These features were selected via an ablation study on a validation set. As evidenced by our results, they yield better accuracies than employing all of them, which causes overfitting. Note that the 2D-3D edges were obtained by projecting the 3D segments onto the 2D superpixels and connecting the pairs of nodes that have a significant projection overlap, i.e., intersection over union more than 0.2).

5.2. Results on NICTA/2D3D

The NICTA/2D3D dataset contains 12 outdoor scenes where each scene is described by a 3D point cloud block together with 10-20 panoramic images. It comprises 14 classes (13 for 3D where *sky* was removed), which yields the following sizes for the parameter matrices: $\mathbf{A}^{2D}_{[14\times23]}$, $\mathbf{A}^{3D}_{[13\times17]}$, $\mathbf{A}^{\Delta}_{[15\times41]}$, $\mathbf{B}^{2D}_{[196\times1]}$, $\mathbf{B}^{3D}_{[169\times1]}$, $\mathbf{B}^{2D-\Delta}_{[210\times1]}$ and $\mathbf{B}^{3D-\Delta}_{[195\times1]}$. The baseline with no latent nodes involves a different parameter matrix of the form $\mathbf{B}^{2D-3D}_{[182\times8]}$ and $\mathbf{B}^{2D-3D}_{[182\times41]}$.

Table 1 and Table 2 compare the results, as F1-scores, of our approach and of the baselines on the 2D and 3D domains, respectively. Note that no results for [15] are available on this dataset. The results in these tables evidence the benefits of using latent nodes, especially on the narrow classes that suffer more from misalignment. On average, our approach clearly outperforms the baselines, and thus achieves the state-of-the-art on this dataset. Furthermore, note that the baseline that utilizes fewer features for the 2D-3D edges is less likely to face overfitting and yields better results.

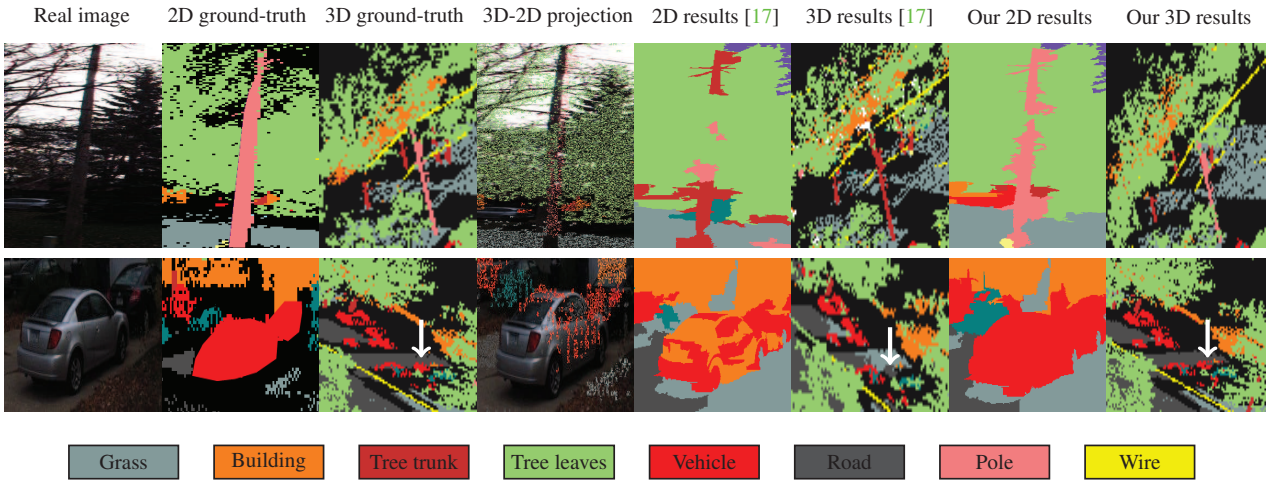

Figure 4. **Examples of how our latent nodes improve the labeling in practice.** As shown in the 3D-2D projection, the data misalignment and object motions have caused 3D points labeled as *leaves* to cover the *pole* (top) and 3D points labeled as *road* to project onto the *vehicles* (bottom). As a consequence, with the method of [17] which encourages the modalities to have the same label, the pole was labeled as *leaves* in 2D and the vehicle as *road* in 3D (indicated by a white arrow). By contrast, thanks to our latent nodes that can cut inconsistent edges, our method produces the correct labels.

In Fig. 4, we illustrate the influence of our latent nodes by two examples. As shown in the figure, cutting the edge between the non-matching 2D and 3D nodes (which have been connected because of misalignment), helps predicting the correct class labels. Fig. 5 shows the results of our approach in one of the scenes in this dataset, compared to the results of [17].

Our results on NICTA/2D3D indicate that, while our latent nodes are in general beneficial, thanks to their ability to cut incorrect connections, they still occasionally yield lower performance than a model without such nodes. We observed that this is mainly due to the inaccurate ground-truth, or to the fact that, sometimes, while the 2D and 3D features appear to be incompatible (e.g., due to challenging viewing conditions), they still belong to the same class. The stronger smoothness imposed by the model without latent nodes is then able to address this issue.

5.3. Results on CMU/VMR

The CMU/VMR dataset is comprised of 372 pairs of urban images and corresponding 3D point cloud data. Importantly, the ground-truth of this data is such that the labels of corresponding 2D and 3D nodes are always the same. In other words, this dataset is not particularly well-suited to our approach. However, it remains a standard benchmark, and no other dataset explicitly evidencing the misalignment problem is available. The CMU/VMR dataset contains 19 classes, which yields parameter matrices of the form $\mathbf{A}^{2D}_{[19\times28]}$, $\mathbf{A}^{3D}_{[19\times23]}$, $\mathbf{A}^{\Delta}_{[20\times52]}$, $\mathbf{B}^{2D}_{[361\times1]}$, $\mathbf{B}^{3D}_{[361\times1]}$, $\mathbf{B}^{2D-\Delta}_{[380\times1]}$ and $\mathbf{B}^{3D-\Delta}_{[380\times1]}$, with alternative matrices for the No Latent baseline of the form $\mathbf{B}^{2D-3D}_{[361\times8]}$ and $\mathbf{B}^{2D-3D}_{[361\times52]}$.

We compare the results of our approach and the baselines on this dataset in Table 3 and Table 4 for the 2D and 3D domains, respectively. In this case, while our approach still yields the best F1-scores on average, there is less difference between our results and the No Latent method. This can easily be explained by the fact that, as mentioned above, the ground-truth labels of corresponding nodes in 2D and 3D are always the same. Furthermore, we can also see that our approach yields low accuracy on classes where few training samples were available, such as the last 5 categories in the tables. This should come at no surprise, since our learning strategy strongly relies on training data. A qualitative comparison is provided in Fig. 6.

6. Conclusion

In this paper, we have addressed the problem of domain inconsistencies in multimodal semantic labeling, which is an important issue when multimodal data is concerned. Such inconsistencies typically cause undesirable connections between 2D and 3D regions, which in turn lead to poor labeling performance. We have therefore proposed a latent CRF model, in which latent nodes supervise the pairwise edges between two domains. Having access to the information of both modalities, these nodes can either improve the labeling in both domains or cut the links between inconsistent 2D and 3D regions. Furthermore, we presented a new set of data-driven learned potentials, which can model complex relationships between the latent nodes and the two domains. Thanks to our latent nodes and our learned potentials, our model achieved state-of-the-art results on two publicly available datasets.

Note that by looking at the data, one can observe that this ground-truth is often wrong, because of the misalignment problem.

Table 1. Per class F1-scores for the 2D domain in the NICTA/2D3D dataset. We compare our model to the method of [17], a pairwise model learned on the 2D domain only, and our model without latent nodes.

	Grass	Building	Tree trunk	Tree leaves	Vehicle	Road	Bush	Pole	Sign	Post	Barrier	Wire	Sidewalk	Sky	avg
Unary	80	33	14	80	49	95	16	28	3	0	0	29	15	98	38
Pairwise 2D (learned)	85	57	17	85	55	95	18	30	0	0	3	34	20	**99**	43
Namin [17]	74	56	21	82	58	92	23	33	**19**	**8**	5	32	29	97	45
No latent (full features)	90	63	10	91	68	96	31	43	1	0	0	**44**	53	**99**	49
No latent (selected features)	92	64	18	92	69	**98**	36	34	3	0	**28**	40	60	**99**	52
Ours	**95**	**71**	**28**	**93**	**76**	97	**44**	**44**	10	5	21	38	**68**	**99**	**56**

Table 2. Per class F1-scores for the 3D domain in the NICTA/2D3D dataset. We compare our model to the method of [17], a pairwise model learned on the 3D domain only, and our model without latent nodes.

	Grass	Building	Tree trunk	Tree leaves	Vehicle	Road	Bush	Pole	Sign	Post	Barrier	Wire	Sidewalk	Sky	avg
Unary	52	61	27	87	58	82	10	24	19	43	19	74	0	#	43
Pairwise 3D (learned)	58	80	**50**	**97**	56	76	16	**62**	32	40	0	**89**	0	#	50
Namin [17]	63	81	41	96	70	76	21	38	28	**47**	23	87	0	#	52
No latent (full features)	**72**	75	27	95	77	**90**	42	**62**	31	9	0	**89**	0	#	52
No latent (selected features)	60	92	45	**97**	75	79	**61**	58	49	29	**27**	82	0	#	58
Ours	66	**94**	49	95	**79**	83	51	**62**	**54**	43	25	**89**	**8**	#	**61**

Table 3. Per class F1-scores for the 2D domain in the CMU/VMR dataset. We compare our model to the method of [15], the method of [17], a pairwise model learned on the 2D domain only, and our model without latent nodes.

	Road	Sidewalk	Ground	Building	Barrier	Bus stop	Stairs	Shrub	Tree trunk	Tree top	Small Vehicle	Big vehicle	Person	Tall light	Post	Sign	Utility pole	Wire	Traffic Signal	avg
Unary	95	81	75	56	29	17	32	50	31	53	32	49	29	**16**	**15**	**16**	33	**41**	29	41
Pairwise 2D (learned)	89	77	74	84	25	17	40	62	37	89	78	57	38	1	5	3	16	12	9	43
Munoz [15]	**96**	**90**	70	83	50	16	33	62	30	86	**84**	50	47	2	9	**16**	14	2	17	45
Namin [17]	94	87	79	74	45	**22**	40	54	27	84	67	24	38	13	2	10	**37**	35	**40**	46
No latent (full features)	93	85	83	**88**	60	4	61	**67**	41	87	79	61	45	0	3	2	12	9	2	46
No latent (selected features)	93	80	80	87	60	1	70	**67**	37	**90**	**84**	**67**	**54**	7	4	4	21	15	3	49
Ours	94	84	**84**	84	**65**	4	**75**	64	**43**	89	**84**	58	52	11	6	2	25	18	3	**50**

Table 4. Per class F1-scores for the 3D domain in the CMU/VMR dataset. We compare our model to the method of [15], the method of [17], a pairwise model learned on the 3D domain only, and our model without latent nodes.

	Road	Sidewalk	Ground	Building	Barrier	Bus stop	Stairs	Shrub	Tree trunk	Tree top	Small Vehicle	Big vehicle	Person	Tall light	Post	Sign	Utility pole	Wire	Traffic Signal	avg
Unary	70	49	62	67	34	2	19	26	11	67	34	4	13	2	0	1	2	0	0	24
Pairwise 3D (learned)	78	52	67	78	15	1	32	31	1	73	44	14	9	1	0	0	0	0	0	26
Munoz [15]	82	73	68	87	46	11	38	63	28	**88**	73	**56**	26	**10**	0	0	0	0	0	39
Namin [17]	**92**	85	81	85	50	**16**	42	55	29	82	70	16	43	6	2	**7**	**29**	9	**23**	43
No latent (full features)	90	86	**87**	**90**	59	2	64	**69**	**31**	79	70	29	47	1	1	0	5	0	0	43
No latent (selected features)	90	85	85	89	62	2	63	68	29	86	**78**	46	**53**	3	1	0	15	0	0	45
Ours	**92**	**88**	84	88	**64**	7	**66**	66	**31**	86	75	42	**53**	8	**7**	0	17	**10**	0	**47**

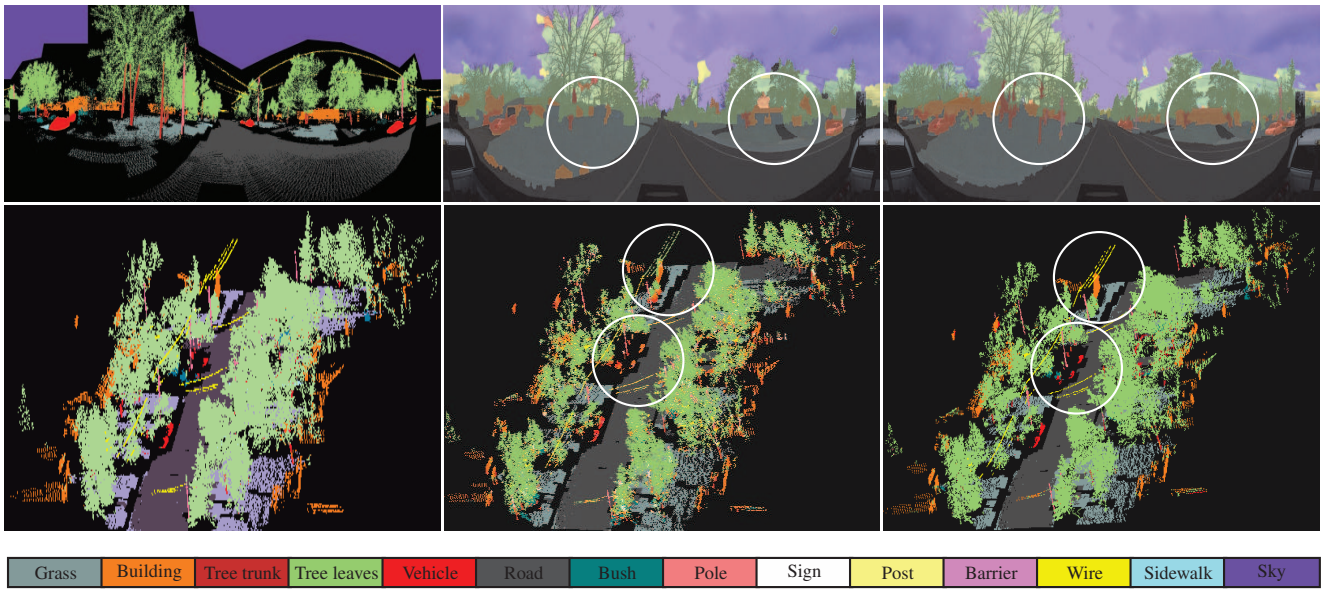

Figure 5. Sample results on the NICTA/2D3D dataset. **1st row: Left:** 2D ground-truth; **Middle:** 2D results of [17]; **Right:** our 2D results. **2nd row: Left:** 3D ground-truth; **Middle:** 3D results of [17]; **Right:** our 3D results. Our method (the right column) has been able to fix some of the mislabelings present in the results of [17], such as the *tree trunks* and *poles* in 2D images, and *wires* and *vehicles* in 3D data. Note that these are the object classes that are most likely to be affected by misalignments.

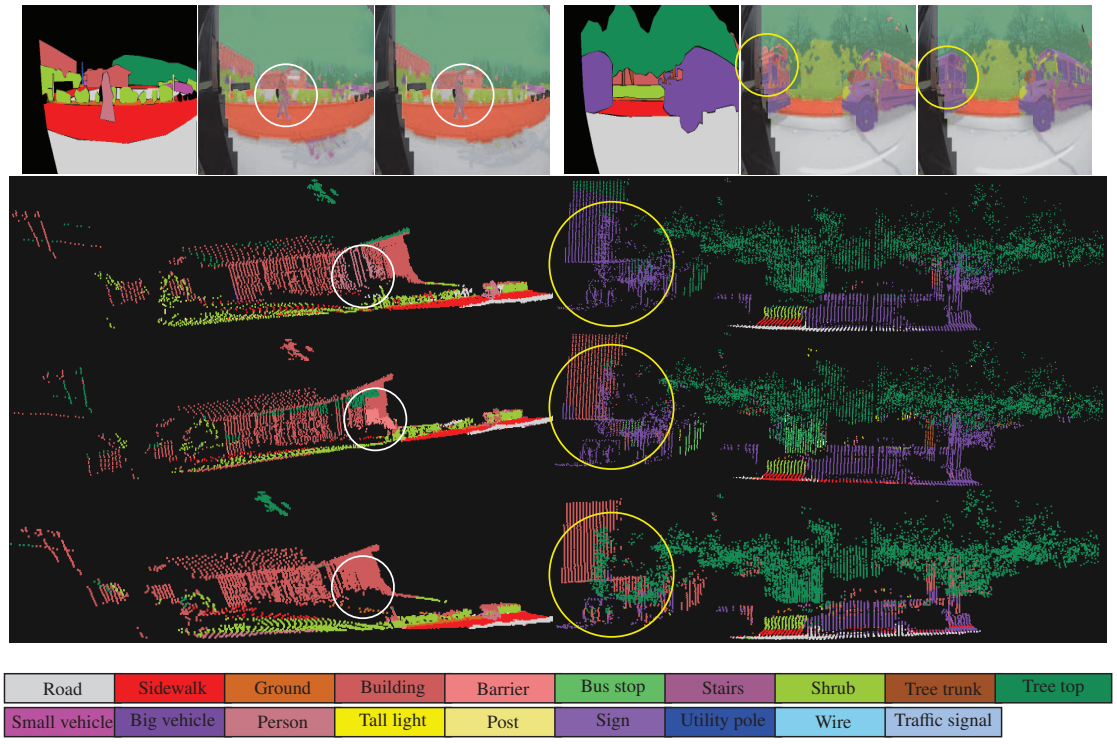

Figure 6. Sample results of two scenes in the CMU/VMR dataset. **1st row in each scene: Left:** 2D ground-truth; **Middle:** the results of [17]; **Right:** our 2D results. **2nd row:** ground-truth of the 3D data; **3rd row:** the results of [17]; **4th row:** our 3D results. The circles highlight mislabeling in the 3D ground-truth of this dataset, which occurred due to misalignments between 2D and 3D data, and illustrate how our method has improved the results in those regions compared to [17].

References

[1] A. Anand, H. Koppula, T. Joachims, and A. Saxena. Contextually guided semantic labeling and search for 3d point clouds. *IJRR*, pages 1729–1736, 2013. 2

[2] C. Cadena and J. Koseck. Semantic Segmentation with Heterogeneous Sensor Coverages. In *ICRA*, 2014. 1, 2

[3] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *PAMI*, 24(5):603–619, 2002. 5

[4] J. Domke. Learning graphical model parameters with approximate marginal inference. *PAMI*, 35(10):2454–2467, 2013. 2, 3, 4, 5

[5] B. Douillard, A. Brooks, and F. Ramos. A 3D laser and vision based classifier. In *ISSNIPC*, 2009. 1, 2

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 2

[7] P. N. Ingmar Posner, Mark Cummins. Fast probabilistic labeling of city maps. In *RSS*, 2008. 1, 2

[8] P. Kohli, L. Ladicky, and P. H. S. Torr. Robust higher order potentials for enforcing label consistency. *IJCV*, 82(3):302–324, 2009. 2

[9] H. Koppula, A. Anand, T. Joachims, and A. Saxena. Semantic labeling of 3D point clouds for indoor scenes. In *NIPS*, 2011. 2

[10] P. Krähenbühl and V. Koltun. Parameter learning and convergent inference for dense random fields. In *ICML*, 2013. 2

[11] L. Ladický, P. Sturgess, K. Alahari, C. Russell, and P. H. S. Torr. What, where and how many? combining object detectors and crfs. ECCV, 2010. 2

[12] A. Levin and Y. Weiss. Learning to combine bottom-up and top-down segmentation. *IJCV*, 81(1):105–118, 2009. 2

[13] D. Lin, S. Fidler, and R. Urtasun. Holistic scene understanding for 3d object detection with rgbd cameras. In *ICCV*, 2013. 2

[14] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, Nov. 2004. 5

[15] D. Munoz, J. A. Bagnell, and M. Hebert. Co-inference for multi-modal scene analysis. In *ECCV*, 2012. 1, 2, 5, 7

[16] M. Najafi, S. T. Namin, M. Salzmann, and L. Petersson. Non-associative higher-order markov networks for point cloud classification. In *ECCV*, 2014. 2

[17] S. T. Namin, M. Najafi, M. Salzmann, and L. Petersson. A multi-modal graphical model for scene analysis. In *WACV*, 2015. 1, 2, 3, 5, 6, 7, 8

[18] X. Ren, C. Fowlkes, and J. Malik. Learning probabilistic models for contour completion in natural images. *IJCV*, 77:47–63, 2008. 2

[19] R. Shapovalov, A. Velizhev, and O. Barinova. Non-associative markov networks for 3d point cloud classification. In *PCVIA*. ISPRS, 2010. 2

[20] G. Singh and J. Kosecka. Nonparametric scene parsing with adaptive feature relevance and semantic context. In *CVPR*, 2013. 2

[21] M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn*, 2008. 4

[22] J. J. Weinman, L. C. Tran, and C. J. Pal. Efficiently learning random fields for stereo vision with sparse message passing. In *ECCV*, 2008. 2

[23] J. Xiao and L. Quan. Multiple view semantic segmentation for street view images. In *ICCV*. IEEE, 2009. 2

[24] J. Yang, B. Price, S. Cohen, and M. Yang. Context driven scene parsing with attention to rare classes. In *CVPR*, 2014. 2

[25] H. Zhang, J. Wang, T. Fang, and L. Quan. Joint segmentation of images and scanned point cloud in large-scale street scenes with low annotation cost. *IEEE TIP*, 2013. 1, 2