Multi-task Learning using Multi-modal Encoder-Decoder Networks with Shared Skip Connections

Ryohei Kuga[1], Asako Kanezaki[2], Masaki Samejima[1], Yusuke Sugano[1], and Yasuyuki Matsushita[1]
[1]Osaka University
[2]National Institute of Advanced Industrial Science and Technology
[1]{kuga.ryohei, samejima, sugano, yasumat}@ist.osaka-u.ac.jp
[2]kanezaki.asako@aist.go.jp

Abstract

Multi-task learning is a promising approach for efficiently and effectively addressing multiple mutually related recognition tasks. Many scene understanding tasks such as semantic segmentation and depth prediction can be framed as cross-modal encoding/decoding, and hence most of the prior work used multi-modal datasets for multi-task learning. However, the inter-modal commonalities, such as one across image, depth, and semantic labels, have not been fully exploited. We propose a multi-modal encoder-decoder networks to harness the multi-modal nature of multi-task scene recognition. In addition to the shared latent representation among encoder-decoder pairs, our model also has shared skip connections from different encoders. By combining these two representation sharing mechanisms, the proposed method efficiently learns a shared feature representation among all modalities in the training data. Experiments using two public datasets shows the advantage of our method over baseline methods that are based on encoder-decoder networks and multi-modal auto-encoders.

1. Introduction

Scene understanding is one of the most important tasks for various applications including robotics and autonomous driving and has been an active research area in computer vision for a long time. The goal of scene understanding can be divided into several different tasks such as depth reconstruction and semantic segmentation. Traditionally, these different tasks have been studied independently using a dedicated methodology. Recently, there is a growing demand for a single unified framework to achieve multiple tasks at a time. By sharing a part of the learned estimator, such a multi-task learning framework is expected to achieve better performance with a compact representation.

In most of the prior work, multi-task learning is formulated with a motivation to train a shared feature representation among different tasks for efficient feature encoding [2, 17, 25]. Accordingly, in recent convolutional neural network (CNN)-based methods, multi-task learning often employs an encoder-decoder network architecture [2, 17, 13]. If, for example, the target tasks are semantic segmentation and depth estimation from RGB images, multi-task networks encode the input image to a shared low-dimensional feature representation and then estimate depth and semantic labels with two distinct decoder networks.

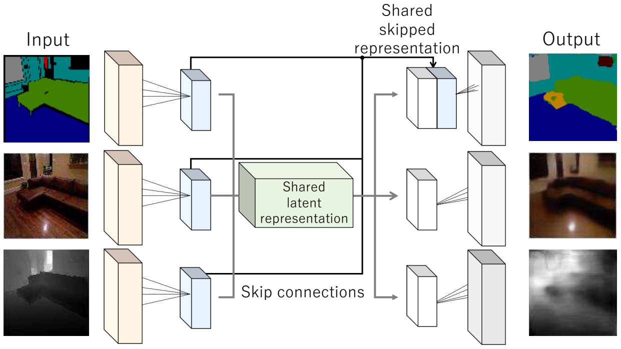

Figure 1. Overview of our multi-modal encoder-decoder networks. Our model takes data in multiple modalities such as RGB images, depth, and semantic labels as input, and generates multi-modal outputs in a multi-task learning framework.

While such a shared encoder architecture can constrain the network to extract a common feature for different tasks, one limitation is that it cannot fully exploit the multi-modal nature of the training dataset. The representation capability of the shared representation in the above example is not limited to image-to-label and image-to-depth conversion tasks, but it can also represent the common feature for *all* of the cross-modal conversion tasks such as depth-to-label as well as within-modal dimensionality reduction tasks such as image-to-image. By incorporating these additional conversion tasks during the training phase, the multi-task network is expected to learn more efficient shared feature representation for the target tasks.

In this work, we propose a multi-modal encoder-decoder networks for multi-task scene recognition. Our model consists of encoders and decoders for each modality, and the whole network is trained in an end-to-end manner taking into account all conversion paths – both cross-modal encoder-decoder pairs and within-modal self-encoders. As illustrated in Fig. 1, all encoder-decoder pairs are connected via one shared latent representation in our method. In addition, inspired by the U-net architecture [20, 12], decoders for pixel-wise image conversion tasks such as semantic segmentation also have shared skipped representations from all encoders. Since the whole network is jointly trained with multi-task losses, these two shared representations are trained to extract the common feature representation among all modalities. Unlike multi-modal auto-encoders [2], our method can further utilize auxiliary unpaired data to train self-encoding paths and consequently improve the cross-modal conversion performance. In the experiments using two public datasets, we show that our proposed architecture performs significantly better on cross-modal conversion tasks.

The contributions of this work are three-fold. First, we propose a novel multi-modal encoder-decoder networks which fully utilizes the multi-modal training data to learn the shared representation among different modalities. Second, we also show that the performance of the proposed method can be further improved by using auxiliary unlabeled data. Finally, through the experimental validation on two public datasets, we show that our method outperforms both of the baseline multi-task network and multi-modal auto-encoders.

2. Related Work

Multi-task learning is motivated by the trait that feature representation for one task could be of use for the other tasks [3]. In prior work, multiple tasks, such as scene classification, semantic segmentation [15], character recognition [30] and depth estimation [8, 7], have been addressed with a single input of an RGB image, which is referred as *Single-modal Multi-task Learning*. Hand *et al*. [10] demonstrated that multi-task learning of gender and facial parts from one face image leads to better accuracy than individual learning of each task. Teichmann *et al*. [26] presented neural networks for scene classification, object detection, segmentation of a street view image. Uhrig *et al*. [28] proposed an instance-level segmentation method via simultaneous estimation of semantic labels, depth, and instance center direction. Li *et al*. [14] proposed fully convolutional neural networks for segmentation and saliency tasks. In those proposed neural networks, feature representation of the single input modality is shared in an intermediate layer for solving multiple tasks. In contrast, the proposed method fully utilizes the multi-modal training data by learning cross-modal shared representations through joint multi-task training.

There have been several prior attempts on utilizing multi-modal inputs for deep neural networks. They proposed to use multi-modal input data such as RGB and depth images [9], visual and textual features [24], audio and video [17], and multiple sensor data [19] for single-task neural networks. In contrast to such multi-modal single-task learning methods, relatively few studies have been made on multi-modal multi-task learning. Ehrlich *et al*. [6] presented a method to identify person's gender and smiling based on two feature modalities extracted from face images. Cadena *et al*. [2] proposed neural networks based on auto-encoder for multi-task estimation of semantic labels and depth.

Both of these single-task and multi-task learning methods with multi-modal data focused on obtaining better shared representation from multi-modal data. Since straightforward concatenation of extracted features from different modalities often results in inaccurate estimation results, some prior methods tried to improve the shared representation by singular value decomposition [1], encoder-decoder [21], auto-encoder [17, 2, 22], and supervised mapping [4]. While our method is also based on the encoder-decoder approach, it employs the U-net architecture for further improving the learned shared representation, particularly in high-resolution convolutional layers.

Most of the prior works also assume that all modalities are available for the single-task or multi-task. One approach for dealing with the missing modal data is zero-filling, which fills zero into the input vector corresponding to the missing modal data [17, 2]. Although these approaches allow the multi-modal networks to handle missing modalities and cross-modal conversion tasks, it has not been fully discussed whether such an zero-filling approach can be also applied to recent CNN-based architectures. Sohn *et al*. [24] explicitly estimated missing modal data from available modal data by deep neural networks. In a difficult task, such as a semantic segmentation with many classes, the missing modal data is estimated inaccurately, which has a negative influence on performance of the whole network. In our method, at the test phase encoder-decoder paths work individually even for missing modal data. Furthermore, our method can perform conversions between all modalities in the training set, and can utilize single-modal data to improve within-modal self-encoding paths.

3. Multi-modal Encoder-Decoder Networks

The architecture of the proposed multi-modal encoder-decoder networks is illustrated in Fig. 2. To exploit the commonality among different tasks, all encoder/decoder pairs are connected with each other via the shared latent repre-

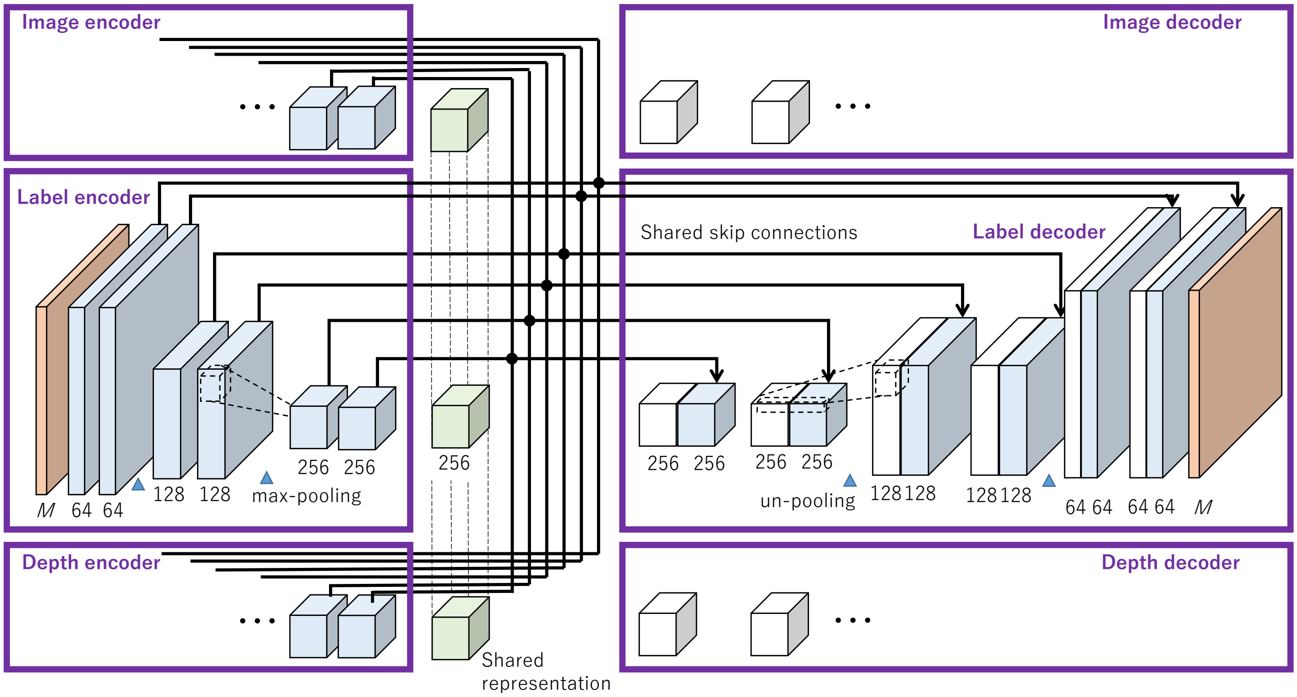

Figure 2. Model architecture of the proposed multi-modal encoder-decoder networks. Our model consists of encoder-decoder networks with the shared latent representation. Depending on the task, the decoder also employs the U-net architecture and connected with all encoders via shared skip connections. The network consists of *Conv+Norm+ReLU* modules except for the final layer, which is equivalent to *Conv*. We use kernel size 3×3 with stride 1 and padding size 1 for all convolutional layer, and kernel size 2×2 and stride 2 for max-pooling.

sentation. In addition, if the decoding task is expected to benefit from high-resolution representations, the decoder is further connected with all encoders via shared skip connections as in the U-net architecture [20]. Given one input modality, the encoder generates a single representation, which is then decoded through different decoders into all available modalities. The whole network is trained by taking into account all combinations of the conversion tasks among different modalities.

In the following, we discuss details of the proposed network by taking the task of depth and semantic label estimation from RGB images assuming a training dataset consisting with three modalities: image, depth and semantic labels. In this example, semantic segmentation is the task where the advantage of the skip connections has been already shown [12], while such high-resolution representations are not always helpful for depth and image decoding tasks. It is also worth noting that the task and the number of modalities are not limited to this particular example. More encoders and decoders can be added to the model, and the decoder can be trained with different tasks and loss functions.

3.1. Model Architecture

As illustrated in Fig. 2, each convolution layer (*Conv*) in the encoder is followed by a batch-normalization layer (*Norm*) and activation function (*ReLU*). Two max-pooling operations are placed in the middle of seven *Conv+Norm+ReLU* components, which makes the dimension of the latent representation 1/16 of the input. Similarly, the decoder network consists of seven *Conv+Norm+ReLU* components except for the final layer, while max-pooling operations are replaced by un-pooling operations for expanding a feature map. The max-pooling operation pools a feature map by taking maximum values, and the un-pooling operation restores the pooled feature into un-pooled feature map by embedding the same values. The final output of the decoder is then rescaled to the original input size. The rescaled output is further fed into a softmax layer to produce the class probability distribution at the label decoder.

As discussed earlier, all encoder/decoder pairs are connected with the shared latent representation. Letting $\mathbf{x} \in \{\mathbf{x}_i, \mathbf{x}_s, \mathbf{x}_d\}$ be the input modalities for each encoder; image, semantic label and depth map, and $\mathbf{E} \in \{\mathbf{E}_i, \mathbf{E}_s, \mathbf{E}_d\}$ be the corresponding encoder functions, then the outputs from each encoder $\mathbf{r}$, which means latent representation, is defined by $\mathbf{r} \in \{\mathbf{E}_i(\mathbf{x}_i), \mathbf{E}_s(\mathbf{x}_s), \mathbf{E}_d(\mathbf{x}_d)\}$. Here, $\mathbf{r} \in \mathbb{R}^{C \times H \times W}$ where C, H, and W are the number of channels, height and width, respectively. Because $\mathbf{r}$ from all the encoders are encoded to the same shape $C \times H \times W$ by the convolution and pooling operations at all $\mathbf{E}$, we can obtain the outputs $\mathbf{y} \in \{\mathbf{D}_i(\mathbf{r}), \mathbf{D}_s(\mathbf{r}), \mathbf{D}_d(\mathbf{r})\}$ from any $\mathbf{r}$, where $\mathbf{D} \in \{\mathbf{D}_i, \mathbf{D}_s, \mathbf{D}_d\}$ are decoder functions. The latent representation between encoders and decoders are not distinguished among different modalities, *i.e.*, the latent representation decoded by an encoder is fed into all decoders, and at the same time each decoder has to be able to decode latent representation from any of the decoders. In other words, latent representation is shared for all encoder/decoder pairs.

For semantic segmentation, we also employ the U-net architecture with skip paths to propagate intermediate low-level features from encoders to decoders. Low-level feature maps in the encoder are concatenated with feature maps generated from latent representations and then convolved in order to mix the features. Since we use 3×3 convolution kernels with 2×2 max pooling operators for the encoder and 3×3 convolution kernels with 2×2 un-pooling operators for the decoder, the encoder and decoder networks are symmetric (U-shape). Our model has skip paths among all combinations of the encoders and decoders, and also shares the low-level features in the same manner as the latent representation.

3.2. Multi-task Training

In the training phase, a batch of training data is passed through all forwarding paths for calculating losses. For example, given a batch of paired RGB, depth, and semantic label images, three decoding losses from the image/depth/label decoders are first computed for the image encoder. The same procedure is then repeated for depth and label encoders, and the global loss is defined as the sum of all decoding losses from nine encoder-decoder pairs. The gradients for the whole network are computed based on the global loss by back-propagation. If the training batch contains unpaired data, we only compute within-modal self-encoding losses. In the following, we describe details of the cost functions defined for semantic label decoder, image decoder, and depth decoder.

Semantic Labels In this work, we define label images so that each pixel has a one-hot-vector that represents the class that the pixel belongs to. The number of the input and output channels is thus equivalent to the number of classes. We define the loss function of semantic label decoding by the pixel-level cross entropy. Letting K be the number of classes, the softmax function is written as

$$p(x)^{(k)} = \frac{\exp\left(f_s(x)^{(k)}\right)}{\sum_{i=1}^{K} \exp\left(f_s(x)^{(i)}\right)}, \tag{1}$$

where $f_s(x)^{(k)} \in \mathbb{R}$ indicates the value at the location x in the k-th channel of the tensor given by the final layer output. Letting P be the whole set of pixels in the output and N be the number of the pixels, the loss function $\mathcal{L}_s$ is defined as

$$\mathcal{L}_s = \frac{1}{N} \sum_{x \in P} \sum_{k=1}^{K} t_k(x) \log p(x)^{(k)}, \tag{2}$$

where $t_k(x) \in \{0, 1\}$ is the k-th channel ground truth label, which is one if the pixel belongs to the k-th class, and zero, otherwise.

RGB Images For image decoding, we set the loss $\mathcal{L}_I$ to the ℓ_1 norm distance of RGB values as

$$\mathcal{L}_I = \frac{1}{N} \sum_{x \in P} \|I(x) - f_i(x)\|_1, \tag{3}$$

where $I(x) \in \mathbb{Z}_+^3$ and $f_i(x) \in \mathbb{R}^3$ are the ground truth and predicted RGB values, respectively. If the goal of the network is realistic RGB image generation from depth and label images, the image decoding loss can be further extended to DCGAN [18] based architectures; however, since the main goal of this work is depth and semantic label prediction, we used the simple ℓ_1 loss for simplicity.

Depth Maps For the depth decoder, we also use ℓ_1 norm distance between the ground truth and predicted depth maps. The loss function $\mathcal{L}_d$ is defined as

$$\mathcal{L}_d = \frac{1}{N} \sum_{x \in P} |d(x) - f_d(x)|, \tag{4}$$

where $d(x) \in \mathbb{R}$ and $f_d(x) \in \mathbb{R}$ are the ground truth and predicted depth values, respectively. We normalize the depth values to $[0, 255]$ by the linear interpolation. In the evaluation step, we revert the normalized depth map into the original scale.

3.3. Implementation Details

In our network, learnable parameters are initialized by a normal distribution. We set the learning rate to 0.001 and the momentum to 0.9 for all layers with weight decay 0.0005. The input image size is fixed to 96×96. We treat the paired and unpaired data as training data, which are mixed randomly in every epoch. Let T denote the number of all RGB images including unpaired data, and $U(\leq T)$ denote the number of RGB images paired with semantic labels and depth maps. A set of training data is denoted by

$\{(I_m, L_n, D_n)\}$, where $1 \leq m \leq T$ and $1 \leq n \leq U$. In the training phase, we prepare a set of RGB, labels, and depth images $\{(I_i, L_i, D_i)\}$ from the paired data ($1 \leq i \leq T$). Training begins with I_i as an input and (L_i, D_i) as outputs. In the next step, L_i is used as an input, and (I_i, D_i) are used as outputs. These steps are repeated on all combinations of the modalities and input/output. A loss is calculated on each combination and used for updating parameters. We train the network by the mini-batch training method with a batch including at least one labeled RGB image.

4. Experiments

In this section, we evaluate the proposed multi-modal encoder-decoder networks for semantic segmentation and depth estimation using two public datasets: NYUDv2 [23] and Cityscape [5]. The baseline model is the single-task encoder-decoder networks (*enc-dec*) and single-modal (RGB image) multi-task encoder-decoder networks (*enc-decs*) that have the same architecture as ours. We also compare our method to multi-modal auto-encoders (*MAE*) [2], which concatenates latent representations of auto-encoders for different modalities. Since the shared representation in MAE is the concatenation of latent representations in all the modalities, it is required to explicitly input zero-filled pseudo signals to estimate the missing modalities. Also, MAE uses fully connected layers instead of convolutional layers, so that input images are flattened when fed into the first layer.

For semantic segmentation, we use the mean intersection over union (MIOU) scores for the evaluation. IOU is defined as

$$IOU = \frac{TP}{TP + FP + FN}, \tag{5}$$

where TP, FP, and FN are the numbers of true positive, false positive, and false negative pixels, respectively, determined over the whole test set. MIOU is the mean of the IOU on the all classes.

For depth estimation, we use several evaluation measures commonly used in prior works [16, 8, 7]:

- Root mean squared error: $\sqrt{\frac{1}{N}\sum_{x \in P}(d(x) - \hat{d}(x))^2}$
- Average relative error: $\frac{1}{N}\sum_{x \in P}\frac{|d(x) - \hat{d}(x)|}{d(x)}$
- Average log 10 error: $\frac{1}{N}\sum_{x \in P}\left|\log_{10}\frac{d(x)}{\hat{d}(x)}\right|$
- Accuracy with threshold:
  Percentage of $x \in P \quad \text{s.t.} \ \max\left(\frac{d(x)}{\hat{d}(x)}, \frac{\hat{d}(x)}{d(x)}\right) < \delta$,

where $d(x)$ and $\hat{d}(x)$ are the ground truth depth and predicted depth at the pixel x, P is the whole set of pixels in an image, and N is the number of the pixels in P.

	Depth Estimation						Semantic Segmentation
	Error			Accuracy			
	Rel	log10	RMSE	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	MIOU
MAE	1.147	0.290	2.311	0.098	0.293	0.491	0.018
enc-dec (U)	-	-	-	-	-	-	0.357
enc-dec	0.340	0.149	1.216	0.396	0.699	0.732	-
enc-decs	0.321	0.150	1.201	0.398	0.687	0.718	0.352
Ours	0.296	0.120	1.046	0.450	0.775	**0.810**	0.411
Ours (+extra)	**0.283**	**0.119**	**1.042**	**0.461**	**0.778**	**0.810**	**0.420**

Table 1. Performance comparison on the NYUDv2 dataset. Each row corresponds to MAE [2], single-task encoder-decoder with and without U-net architecture, single-modal multi-task encoder-decoder, our method with and without extra RGB training data. First six columns show performance metrics for depth estimation, and the last column shows semantic segmentation performances.

4.1. Results on NYUDv2 dataset

NYUDv2 dataset has $1,448$ images annotated with semantic labels and measured depth values. This dataset contains 868 images for training and a set of 580 images for testing. We divided test data into 290 and 290 for validation and testing respectively, and used the validation data for early stopping. The dataset also has extra $407,024$ unpaired RGB images, and we randomly selected $10,459$ images as unpaired training data, while *other* class was not considered in both of training and evaluation. For semantic segmentation, following prior work [11, 29, 7], we evaluate the performance on estimating 12 classes out of all available classes.

Table 1 shows depth estimation and semantic segmentation results of all methods. Each row corresponds to MAE [2], single-task encoder-decoder with and without skip connections, single-modal multi-task encoder-decoder, our method with and without extra RGB training data. First six columns show performance metrics for depth estimation, and the last column shows semantic segmentation performances. The performance of our method was better than the single-task network (enc-dec) and single-modal multi-task encoder-decoder network (enc-decs) on all metrics even without the extra data, showing the effectiveness of the multi-modal architecture. The performance was further improved with the extra data, and achieved the best performance in all evaluation metrics. It shows the benefit of using unpaired training data and multiple modalities to learn more effective representations. More detailed results on semantic segmentation are shown in Table 2. Each column shows class-specific IOU scores for all models. Our model with extra training data outperforms the baseline models with 10 out of the 12 classes and achieved 0.063 points improvement in MIOU.

	book	cabinet	ceiling	floor	table	wall	window	picture	blinds	sofa	bed	tv	mean
MAE	0.002	0.033	0.000	0.101	0.020	0.023	0.022	0.005	0.001	0.004	0.006	0.000	0.018
enc-dec (U)	0.055	0.371	0.472	0.648	0.197	0.711	0.334	0.361	0.274	0.302	0.370	0.192	0.357
enc-decs	0.071	0.382	0.414	0.659	0.222	0.706	**0.363**	0.336	0.234	0.300	0.320	0.220	0.352
Ours	**0.096**	0.480	0.529	0.704	0.237	0.745	0.321	0.414	0.303	0.365	**0.455**	0.285	0.411
Ours (+extra)	0.072	**0.507**	**0.534**	**0.736**	**0.299**	**0.749**	0.320	**0.422**	**0.304**	**0.375**	0.413	**0.307**	**0.420**

Table 2. Detailed IOU on the NYUDv2 dataset. Each column shows class-specific IOU scores for all models.

	road	side walk	build ing	wall	fence	pole	traffic light	traffic sigh	vege tation	terrain	sky	person	rider	car	truck	bus	train	motor cycle	bicycle	mean
MAE	0.688	0.159	0.372	0.022	0.000	0.000	0.000	0.000	0.200	0.000	0.295	0.000	0.000	0.137	0.000	0.000	0.000	0.000	0.000	0.099
enc-dec	0.931	0.556	0.757	0.125	0.054	0.230	0.100	0.164	0.802	0.430	0.869	0.309	0.040	0.724	0.062	0.096	0.006	0.048	0.270	0.346
enc-decs	0.936	0.551	0.769	0.128	0.051	0.220	0.074	0.203	0.805	0.446	0.887	0.318	**0.058**	0.743	0.051	0.152	0.077	0.056	0.241	0.356
Ours	0.925	0.529	0.770	0.053	0.036	0.225	0.049	0.189	0.805	0.445	0.867	0.325	0.007	0.720	**0.075**	0.153	**0.133**	0.043	0.218	0.346
Ours (+extra)	**0.950**	**0.640**	**0.793**	**0.172**	**0.062**	**0.280**	**0.109**	**0.231**	**0.826**	**0.498**	**0.890**	**0.365**	0.036	**0.788**	0.035	**0.251**	0.032	**0.108**	**0.329**	**0.389**

Table 4. Detailed IOU on the Cityscape dataset.

	Depth Estimation						Semantic Segmentation
	Error			Accuracy			
	Rel	log10	RMSE	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	MIOU
MAE	3.675	0.441	34.583	0.213	0.395	0.471	0.099
enc-dec (U)	-	-	-	-	-	-	0.346
enc-dec	0.380	0.125	8.983	0.602	0.780	0.870	-
enc-decs	0.365	0.117	8.863	0.625	0.798	0.880	0.356
Ours	0.387	0.115	8.267	0.631	0.803	0.887	0.346
Ours (+extra)	**0.290**	**0.100**	**7.759**	**0.667**	**0.837**	**0.908**	**0.389**

Table 3. Performance comparison on the Cityscape dataset.

4.2. Results on Cityscape dataset

The Cityscape dataset consists of $2,975$ images for training and 500 images for validation, which are provided together with semantic labels and disparity. We divide the validation data into 250 and 250 for validation and testing respectively, and used the validation data for early stopping. This dataset has $19,998$ additional RGB images without annotations, and we also used them as extra training data. There are semantic labels of 19 class objects and a single background (unannotated) class. We used the 19 classes (excluding the background class) for evaluation. For depth estimation, we used the disparity maps provided together with the dataset as the ground-truth. Since there were missing disparity values in the raw data unlike NYUDv2, we adopted the image inpainting method [27] to interpolate disparity maps for both training and testing.

The results are shown in Table 3, and the detailed comparison on semantic segmentation are summarized in Table 4. Our model achieved improvement over both of the MAE [2] and our baseline networks in most of the target classes. While the proposed method without extra data did not improve MIOU, it resulted in 0.043 points improvement with extra data. Our method also achieved the best performance on the depth estimation task, and the performance gain from extra data illustrates the generalization capability of the proposed training strategy.

	Depth Estimation						Semantic Segmentation
	Error			Accuracy			
	Rel	log10	RMSE	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	MIOU
image-to-depth	0.283	0.119	1.042	0.461	0.778	0.810	-
label-to-depth	0.258	0.128	1.114	0.452	0.741	0.779	-
image-to-label	-	-	-	-	-	-	0.420
depth-to-label	-	-	-	-	-	-	0.476

Table 5. Comparison of auxiliary task performances on the NYUDv2.

4.3. Auxiliary Tasks

Although our main goal is semantic segmentation and depth estimation from RGB images, in Fig. 3 we show other cross-modal conversion pairs, *i.e.*, semantic segmentation from depth images and depth estimation from semantic labels on cityscape dataset. From left to right, each column corresponds to ground-truth (a) RGB image, (b) upper for semantic label image lower for depth map and (c) upper for image-to-label lower for image-to-depth, (d) upper for depth-to-label lower for label-to-depth. The ground truth depth maps are ones after inpainting. As can be seen, our model could also reasonably perform these auxiliary tasks.

More detailed examples and evaluations on NYUDv2 dataset is shown in Fig. 4 and Table 5. Left side of Fig. 4 shows examples of output images corresponding to all of the above-mentioned tasks. From top to bottom on the left side, each row corresponds to the ground-truth (a) RGB image, (b) semantic label image, estimated semantic labels

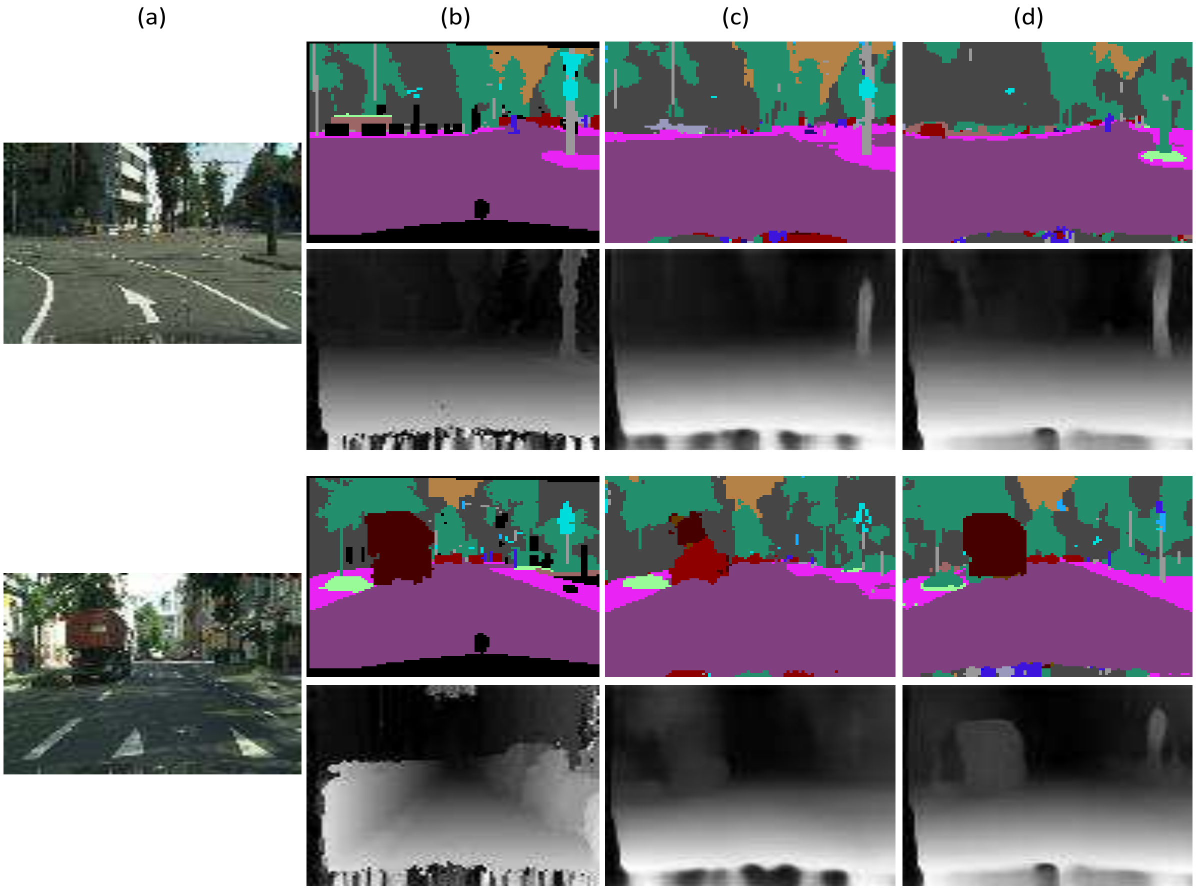

Figure 3. Example output images from our model on the Cityscape dataset. From left to right, each column corresponds to (a) input RGB image, (b) the ground-truth semantic label image (top) and depth map (bottom), (c) estimated label and depth from RGB image, (d) estimated label from depth image (top) and estimated depth from label image (bottom).

from (c) the baseline enc-dec model, (d) image-to-label, (e) depth-to-label conversion paths of our method. (f) depth map (normalized to [0,255] for visualization) and estimated depth maps from (g) enc-dec, (h) image-to-depth, (i) label-to-depth. Interestingly, these auxiliary tasks achieved better performances than the RGB input cases. Clearer object boundary in the label and depth images is one of the potential reasons of the performance improvement. In addition, right side of Fig. 4 shows image decoding tasks and each block corresponds to (a) the ground-truth RGB image, (b) semantic label, (c) depth map, (d) label-to-image, and (e) depth-to-image. Although our model could not correctly reconstruct the input color, object shapes can be seen even with the simple image reconstruction loss.

5. Conclusion

This paper proposed a multi-modal encoder-decoder networks for efficient multi-task learning with a shared feature representation. In our method, encoders and decoders are connected via the shared latent representation and shared skipped representations. Experiments showed the potential of shared representations from different modalities to improve the multi-task performance.

One of the most important future works is to investigate the effectiveness of the proposed multi-modal encoder-decoder networks on different tasks such as image captioning and DCGAN-based image translation. More detailed investigation on learned shared representations during multi-task training is another important future direction to understand why and how the multi-modal encoder-decoder architecture addresses the multi-modal conversion tasks.

Acknowledgments

This project is supported by the New Energy and Industrial Technology Development Organization (NEDO).

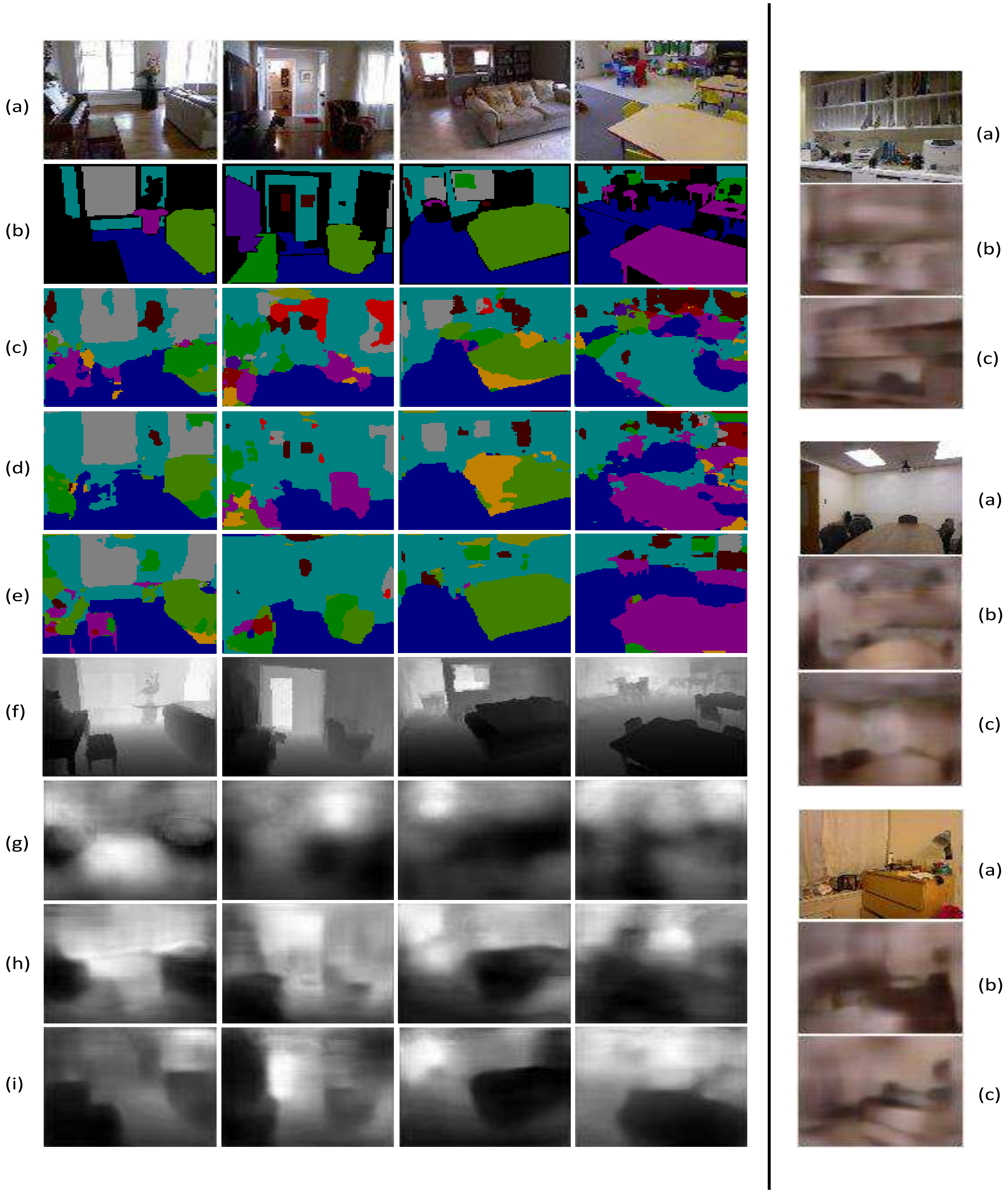

Figure 4. Example outputs from our model on the NYUDv2 dataset. From top to bottom on the left side, each row corresponds to (a) the input RGB image, (b) the ground-truth semantic label image, (c) estimation by enc-dec, (d) image-to-label estimation by our method, (e) depth-to-label estimation by our method, (f) estimated depth map by our method (normalized to $[0, 255]$ for visualization), (g) estimation by enc-dec, (h) image-to-depth estimation by our method, and (i) label-to-depth estimation by our method. In addition, the right side shows image decoding tasks, where each block corresponds to (a) the ground-truth RGB image, (b) label-to-image estimate, and (c) depth-to-image estimate.

References

[1] E. Bruni, N. K. Tran, and M. Baroni. Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, 49(1):1–47, 2014. 2

[2] C. Cadena, A. Dick, and I. D. Reid. Multi-modal auto-encoders as joint estimators for robotics scene understanding. In *Proceedings of Robotics: Science and Systems*, 2016. 1, 2, 5, 6

[3] R. Caruana. Multitask learning. *Machine Learning*, 28(1), 1997. 2

[4] G. Collell, T. Zhang, and M. Moens. Imagined visual representations as multimodal embeddings. In *Proceedings of AAAI Conference on Artificial Intelligence*, 2017. 2

[5] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 5

[6] M. Ehrlich, T. J. Shields, T. Almaev, and M. R. Amer. Facial attributes classification using multi-task representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016. 2

[7] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015. 2, 5

[8] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Proceedings of Advances in neural information processing systems*, 2014. 2, 5

[9] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard. Multimodal deep learning for robust RGB-D object recognition. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2015. 2

[10] E. M. Hand and R. Chellappa. Attributes for improved attributes: A multi-task network for attribute classification. *arXiv preprint arXiv:1604.07360*, 2016. 2

[11] A. Hermans, G. Floros, and B. Leibe. Dense 3D semantic mapping of indoor scenes from RGB-D images. In *Proceedings of IEEE International Conference on Robotics and Automation*, 2014. 5

[12] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2, 3

[13] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. *arXiv preprint arXiv:1705.07115*, 2017. 1

[14] X. Li, L. Zhao, L. Wei, M. H. Yang, F. Wu, Y. Zhuang, H. Ling, and J. Wang. DeepSaliency: Multi-task deep neural network model for salient object detection. *IEEE Transactions on Image Processing*, 25(8):3919–3930, 2016. 2

[15] Y. Liao, S. Kodagoda, Y. Wang, L. Shi, and Y. Liu. Understand scene categories by objects: A semantic regularized scene classifier using convolutional neural networks. In *Proceedings of IEEE International Conference on Robotics and Automation*, 2016. 2

[16] F. Liu, C. Shen, and G. Lin. Deep convolutional neural fields for depth estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015. 5

[17] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning*, 2011. 1, 2

[18] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 4

[19] V. Radu, N. D. Lane, S. Bhattacharya, C. Mascolo, M. K. Marina, and F. Kawsar. Towards multimodal deep learning for activity recognition on mobile devices. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, pages 185–188, 2016. 2

[20] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2015. 2, 3

[21] I. V. Serban, A. G. Ororbia II, J. Pineau, and A. C. Courville. Multi-modal variational encoder-decoders. *http://arxiv.org/abs/1612.00377*, 2016. 2

[22] C. Silberer, V. Ferrari, and M. Lapata. Visually grounded meaning representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1–1, 2016. 2

[23] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *Proceedings of European Conference on Computer Vision*, pages 746–760, 2012. 5

[24] K. Sohn, W. Shang, and H. Lee. Improved multimodal deep learning with variation of information. In *Advances in Neural Information Processing Systems*, pages 2141–2149. 2014. 2

[25] N. Srivastava and R. R. Salakhutdinov. Multimodal learning with deep boltzmann machines. In *Proceedings of Advances in neural information processing systems*, 2012. 1

[26] M. Teichmann, M. Weber, M. Zoellner, R. Cipolla, and R. Urtasun. Multinet: Real-time joint semantic reasoning for autonomous driving. *arXiv preprint arXiv:1612.07695*, 2016. 2

[27] A. Telea. An image inpainting technique based on the fast marching method. *Journal of graphics tools*, 9(1):23–34, 2004. 6

[28] J. Uhrig, M. Cordts, U. Franke, and T. Brox. Pixel-level encoding and depth layering for instance-level semantic labeling. In *Proceedings of German Conference on Pattern Recognition*, 2016. 2

[29] A. Wang, J. Lu, G. Wang, J. Cai, and T.-J. Cham. Multimodal unsupervised feature learning for RGB-D scene labeling. In *Proceedings of European Conference on Computer Vision*, 2014. 5

[30] Y. Yang and T. Hospedales. Deep multi-task representation learning: A tensor factorisation approach. In *Proceedings of 5th International Conference on Learning Representations*, 2017. 2