Learning Task-Specific and Shared Representations in Medical Imaging

Felix J.S. Bragman$^{\star 1,2}$, Ryutaro Tanno$^{\star 1,3,4}$
Sebastien Ourselin^{2}, Daniel C. Alexander^{1,3} and M. Jorge Cardoso^{2,1}

[1] Centre for Medical Image Computing, University College London, United Kingdom
[2] Artificial Medical Intelligence Group, Biomedical Engineering and Imaging Sciences
King's College London, United Kingdom
[3] Department of Computer Science, University College London, United Kingdom
[4] Machine Intelligence and Perception Group, Microsoft Research Cambridge, United Kingdom

Abstract. The performance of multi-task learning hinges on the design of feature sharing between tasks; a process which is combinatorial in the network depth and task count. Hand-crafting an architecture based on human intuitions of task relationships is therefore suboptimal. In this paper, we present a probabilistic approach to learning task-specific and shared representations in Convolutional Neural Networks (CNNs) for multi-task learning of semantic tasks. We introduce Stochastic Filter Groups; which is a mechanism that groups convolutional kernels into task-specific and shared groups to learn an optimal kernel allocation. They facilitate learning optimal shared and task specific representations. We employ variational inference to learn the posterior distribution over the possible grouping of kernels and CNN weights. Experiments on MRI-based prostate radiotherapy organ segmentation and CT synthesis demonstrate that the proposed method learns optimal task allocations that are inline with human-optimised networks whilst improving performance over competing baselines.

1 Introduction

The performance of predictive models is tied to the quality of the learned representations. This is important in medical image computing; where the learned low-dimensional embeddings [1] or features representing the spectrum of disease phenotypes [2] influence the utility of automated clinical tools. Multi-task learning (MTL) has been successful in medical image analysis [3, 4] as it can enhance learning efficiency and model performance by leveraging the inductive bias when jointly solving related tasks. [5]

A key factor for successful MTL models is the ability to determine when to share features within a network. A mechanism is needed to understand the commonalities and differences between tasks to effectively transfer information while optimising weights for individual tasks. The quality of this process is determined by the architectural design, where features or weights are either shared or task specific [6, 7]. However, the space of possible architectures is combinatorially large whilst manual exploration of this space is inefficient and subject to bias on prior beliefs of task relationships. The

[⋆] Both authors contributed equally

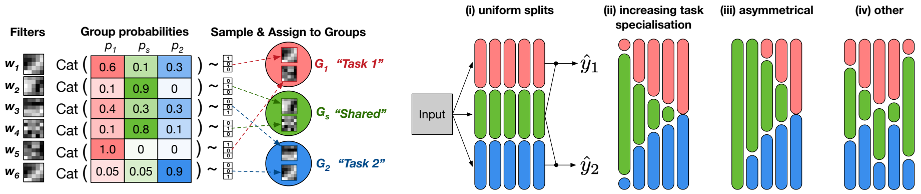
Filters
Group probabilities
Sample & Assign to Groups
p_1
p_s
p_2
w_1 Cat (0.6 0.1 0.3) ~
w_2 Cat (0.1 0.9 0) ~
w_3 Cat (0.4 0.3 0.3) ~
w_4 Cat (0.1 0.8 0.1) ~
w_5 Cat (1.0 0 0) ~
w_6 Cat (0.05 0.05 0.9) ~
G_1 "Task 1"
G_s "Shared"
G_2 "Task 2"
(i) uniform splits
(ii) increasing task specialisation
(iii) asymmetrical
(iv) other
Input
$\hat{y}_1$
$\hat{y}_2$

Fig. 1: Left) Illustration of filter assignment in a SFG module. Right) Possible grouping patterns learnable with the proposed method. The pink, green and blue blocks represent the ratio of filter groups G_1 (pink), G_s (green) and G_2 (blue), where (i) is the case a uniform kernel split, (ii) & (iii) where the networks becomes increasingly task-specific (iv) and an heterogeneous task split.

number of kernels to allocate to each task or to a joint representation depends on the difficulty of individual tasks and the relationship between them [8]; neither of which are *a priori* known in most cases.

In an MTL setting, one would like to learn where to share network components across tasks to maximise performance. The main challenge lies in designing a mechanism that determines how and where to share CNN weights. There are broadly two categories for weight sharing in MTL networks. The first directly optimises weight sharing to maximise task-wise performance by learning a set a vectors that control which features are shared within a layer and how these are distributed across [6, 7, 9]. The second group of MTL methods focuses on weight clustering based on task-similarity, which can be performed by iteratively growing a tree-like deep architecture that clusters similar tasks hierarchically [10] or through maximising task's statistical dependency [11].

In this paper, we propose *Stochastic Filter Groups* (SFGs); a probabilistic mechanism that learns how to allocate kernels to task-specific and shared groups in each layer of MTL architectures (Fig. 1-Right). Specifically, the SFGs learn to allocate kernels in each convolutional layer into either "specialist" groups or a "shared" trunk, which are respectively specific to or shared across different tasks. The SFG equips the network with a mechanism to learn inter-layer connectivity and thus the structure of task-specific and shared representations. We evaluate the efficacy of SFGs on a joint semantic regression (i.e. image synthesis) and semantic segmentation (i.e. organ segmentation) problem applied to prostate data. Experiments show the proposed method achieves higher prediction accuracy than baselines with no mechanism to learn connectivity structures. Importantly, we also demonstrate that the learned representations are meaningful and specific towards each task.

2 Methods

We introduce a new approach for learning learn task-specific and shared representations in multi-task CNN architectures applied to medical imaging tasks i.e. modality transfer and organ segmentation. We propose *stochastic filter groups* (SFG), a probabilistic mechanism to partition kernels in each CNN layer into "specialist" groups or a "shared" group, which are specific to or shared across different tasks, respectively. We use variational inference to learn the distributions over the possible grouping of kernels and network weights that determines connectivity between layers and the shared and task-specific features. This naturally results in a learning algorithm that optimally allocates representation capacity across multi-tasks via gradient-based stochastic optimization.

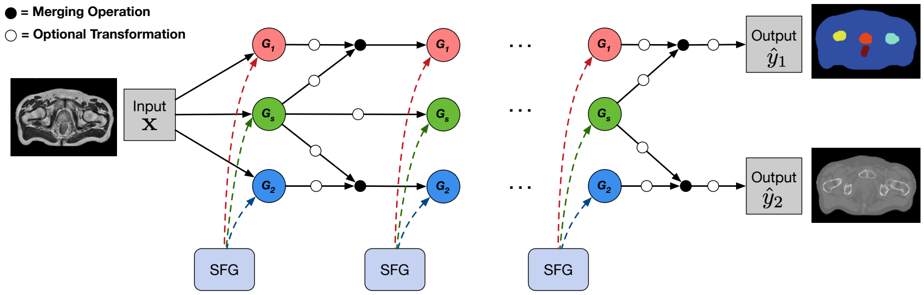

Fig. 2: Multi-task architecture based on SFG modules, where at each layer, kernels are stochastically assigned to task-specific and shared groups.

We consider the synthesis of a CT scan from MRI whilst simultaneously segmenting organs (Fig. 2). This is a significant challenge in MR-only radiotherapy treatment planning, which is attempting to eliminate CT acquisition for treatment planning. This is a complex task that can benefit from multi-task learning and disentangling anatomical rep)resentations since CT synthesis can exploit context from the segmentation whilst there are features specific to CT synthesis not necessarily useful for organ segmentation.

2.1 Stochastic Filter Groups

SFGs introduce a sparse connection structure into the architecture of CNN for multi-task learning to separate features into task-specific and shared components. Ioannou et al. [12] introduced *filter groups* to partition kernels in each convolution layer into groups, each of which acts only on a subset of the preceding features demonstrating that such sparsity reduces computational cost without compromising accuracy. We adapt the concept of filter groups to MTL and propose a mechanism for learning an optimal kernel grouping rather than pre-specifying them.

For simplicity, we describe SFGs for the case of two semantic tasks; image regression and object segmentation. At the l^{th} convolution layer in a CNN architecture with K_l kernels $\{\mathbf{w}^{(l),k}\}_{k=1}^{K_l}$, the associated SFG performs two operations:

1 - Filter Assignment: each kernel $\mathbf{w}_k^{(l)}$ is stochastically assigned to either: i) the "regression specific group" $G_{reg}^{(l)}$, ii) "shared group" $G_s^{(l)}$ or iii) "segmentation specific group" $G_{seg}^{(l)}$ with respective probabilities $\mathbf{p}^{(l),k} = [p_{reg}^{(l),k}, p_s^{(l),k}, p_{seg}^{(l),k}] \in [0,1]^3$. Convolving with the respective filter groups yields distinct sets of features $F_{reg}^{(l)}, F_s^{(l)}, F_{seg}^{(l)}$. Fig. 1-Left illustrates this operation and Fig. 1-Right shows different learnable patterns.

2 - Feature Routing: the features $F_{reg}^{(l)}, F_s^{(l)}, F_{seg}^{(l)}$ are routed to the filter groups $G_{reg}^{(l+1)}$, $G_s^{(l+1)}, G_{seg}^{(l+1)}$ in the subsequent $(l+1)^{\text{th}}$ layer to respect the task-specificity and sharing of filter groups in the l^{th} layer. Specifically, we perform the following routing for $l > 0$ where $F_{reg}^{(l+1)} = h^{(l+1)}\big([F_{reg}^{(l)}|F_s^{(l)}] * G_{reg}^{(l+1)}\big)$, $F_s^{(l+1)} = h^{(l+1)}\big(F_s^{(l)} * G_s^{(l+1)}\big)$, and $F_{seg}^{(l+1)} = h^{(l+1)}\big([F_{seg}^{(l)}|F_s^{(l)}] * G_{seg}^{(l+1)}\big)$ and each $h^{(l+1)}$ defines the non-linear function, $*$ denotes convolution operation and $|$ denotes a merging operation of arrays (e.g. concatenation). At $l = 0$, input image $\mathbf{x}$ is simply convolved with the first set of filter groups to yield $F_i^{(1)} = h^{(1)}\big(\mathbf{x} * G_i^{(1)}\big), i \in \{reg, seg, s\}$.

Fig. 2 provides a schematic of our architecture, in which each SFG module stochastically generates filter groups in each layer and the resultant features are sparsely routed as described above. The merging modules, denoted as black circles, combine the task-specific and shared features appropriately, i.e. $[F_i^{(l)}|F_s^{(l)}], i = reg, seg$ and pass them to the filter groups in the next layer. Each white circle denotes the presence of additional transformations (e.g. convolutions or fully connected layers) in each $h^{(l+1)}$, performed on top of the standard non-linearity (e.g. ReLU).

The proposed sparse connectivity is integral to ensure task performance and structured representations. In particular, one might argue that routing of "shared" features $F_s^{(l)}$ to the respective "task-specific" filter groups $G_{reg}^{(l+1)}$ and $G_{seg}^{(l+1)}$ is not necessary to ensure the separation of gradients across the task losses. However, this connection allows for learning more complex task-specific features at deeper layers in the network.

The varying dimensionality of feature maps is noteworthy. Specifically, the number of kernels in the respective filter groups $G_{reg}^{(l)}, G_s^{(l)}, G_{seg}^{(l)}$ can vary at each iteration of training, which influences the depth of the resultant feature maps $F_{reg}^{(l)}, F_s^{(l)}, F_{seg}^{(l)}$. To work with features maps of varying size, we implement the proposed architecture by defining $F_{reg}^{(l)}, F_s^{(l)}, F_{seg}^{(l)}$ as sparse tensors. At each SFG module, we first convolve the input features with all kernels, and generate the output features from each filter group by zeroing out the channels that root from the kernels in the other groups, resulting in $F_{reg}^{(l)}, F_s^{(l)}, F_{seg}^{(l)}$ that are sparse at non-overlapping channel indices. In the simplest form with no additional transformation (i.e. the grey circles in Fig. 2 are identity functions), we define the merging operation $[F_i^{(l)}|F_s^{(l)}], i = reg, seg$ as pixel-wise summation. In the presence of more complex transforms (e.g. residual blocks), we concatenate the output features in the channel-axis and perform a 1x1 convolution to ensure the number of channels in $[F_i^{(l)}|F_s^{(l)}]$ is the same as in $F_s^{(l)}$.

2.2 T+1 Way "Drop-Out"

The CNN weights and grouping probabilities are simultaneously optimised by extending the variational interpretation of binary dropout [13] to the $(T+1)$-way assignment of each convolution kernel to the filter groups where T is the number of tasks. We consider the case $T = 2$ for CT synthesis and organ segmentation.

Suppose that the architecture consists of L SFG modules, each with K_l kernels where l is the index. As the posterior distribution over the convolution kernels in SFG modules $p(\mathcal{W}|\mathbf{X}, \mathbf{Y}^{(reg)}, \mathbf{Y}^{(seg)})$ is intractable, we approximate it with a simpler distribution $q_\phi(\mathcal{W})$ where $\mathcal{W} = \{\mathbf{w}^{(l),k}\}_{k=1,\ldots,K_l,l=1,\ldots,L}$. Assuming that the posterior distribution factorizes over layers and kernels up to group assignment, we defined the variational distribution as:

$$q_{\phi}(\mathcal{W}) = \prod_{l=1}^{L}\prod_{k=1}^{K_l} q_{\phi_{lk}}(\mathbf{w}^{(l),k}) = \prod_{l=1}^{L}\prod_{k=1}^{K_l} q_{\phi_{lk}}(\mathbf{w}_{reg}^{(l),k}, \mathbf{w}_{s}^{(l),k}, \mathbf{w}_{seg}^{(l),k})$$

where $\{\mathbf{w}_{reg}^{(l),k}, \mathbf{w}_{s}^{(l),k}, \mathbf{w}_{seg}^{(l),k}\}$ denotes the k^{th} kernel in l^{th} layer after routing into task-specific $G_{reg}^{(l)}, G_{seg}^{(l)}$ and shared $G_{s}^{(l)}$ groups. Each $q_{\phi_{lk}}(\mathbf{w}_{reg}^{(l),k}, \mathbf{w}_{seg}^{(l),k}, \mathbf{w}_{s}^{(l),k})$ is defined as $\mathbf{w}_{i}^{(l),k} = z_{i}^{(l),k} \cdot \mathbf{w}^{(l),k}$ for $i \in \{reg, s, seg\}$, where $\mathbf{z}^{(l),k} = [z_{reg}^{(l),k}, z_{seg}^{(l),k}, z_{s}^{(l),k}] \sim$ Cat$(\mathbf{p}^{(l),k})$. Here, $\mathbf{z}^{(l),k}$ is the sample one-hot encoding from the Categorical distribution over filter group assignments. The variational parameters ϕ_{lk} consists of pre-grouping convolution kernels $\mathbf{w}^{(l),k}$ and grouping probabilities $\mathbf{p}^{(l),k} = [p_{reg}^{(l),k}, p_{s}^{(l),k}, p_{seg}^{(l),k}]$.

We minimize the KL divergence between the approximate posterior $q_{\phi}(\mathcal{W})$ and $p(\mathcal{W}|\mathbf{X}, \mathbf{Y}^{(reg)}, \mathbf{Y}^{(seg)})$. Assuming likelihood factorisation over the two tasks, we have the following objective $\mathcal{L}_{\text{MC}}(\phi) = -\frac{N}{M}\sum_{i=1}^{M}\Big[\log p(y_i^{(reg)}|\mathbf{x}_i, \mathcal{W}_i) + \log p(y_i^{(seg)}|\mathbf{x}_i, \mathcal{W}_i)\Big] + \sum_{l=1}^{L}\sum_{k=1}^{K_l} \text{KL}\big(q_{\phi_{lk}}(\mathbf{w}^{(l),k})||p(\mathbf{w}^{(l),k})\big)$, where M is the size of the mini-batch, N is the total number of training data points, and $\mathcal{W}_i$ denotes a set of model parameters sampled from $q_{\phi}(\mathcal{W})$. The last KL term regularizes the deviation of the approximate posterior from the prior $p(\mathbf{w}^{(l),k}) = \mathcal{N}(0, \mathbf{I}/l^2)$ where $l > 0$. Adapting the approximation presented in [13] to our scenario, we obtain:

$$\text{KL}(q_{\phi_{lk}}(\mathbf{w}^{(l),k})||p(\mathbf{w}^{(l),k})) \propto \frac{l^2}{2}||\mathbf{w}^{(l),k}||_2^2 - \mathcal{H}(\mathbf{p}^{(l),k}) \tag{1}$$

where $\mathcal{H}(\mathbf{p}^{(l),k}) = -\sum_{i\in\{reg,seg,s\}} p_i^{(l),k} \log p_i^{(l),k}$ is the entropy of grouping probabilities. The first term performs the L2-weight norm and the second term pulls the grouping probabilities towards the uniform distribution. The overall loss is defined as:

$$\mathcal{L}_{\text{MC}}(\phi) = -\frac{N}{M}\sum_{i=1}^{M}\Big[\log p\Big(y_i^{(1)}|\mathbf{x}_i, \mathcal{W}_i\Big) + \log p\Big(y_i^{(2)}|\mathbf{x}_i, \mathcal{W}_i\Big)\Big] \\ + \lambda_1 \cdot \sum_{l=1}^{L}\sum_{k=1}^{K_l} ||\mathbf{w}^{(l),k}||_2^2 - \lambda_2 \cdot \sum_{l=1}^{L}\sum_{k=1}^{K_l} \mathcal{H}(\mathbf{p}^{(l),k}) \tag{2}$$

where $\lambda_1 > 0, \lambda_2 > 0$ are regularization coefficients.

The discrete sampling operation during filter group assignment creates discontinuities, giving the first term in the objective function (eq. 2) zero gradient with respect to the grouping probabilities $\{\mathbf{p}^{(l),k}\}$. We approximate each of the categorical variables Cat$(\mathbf{p}^{(l),k})$ by the Gumbel-Softmax distribution, GSM$(\mathbf{p}^{(l),k}, \tau)$ [14], a continuous relaxation which allows for sampling, differentiable with respect to the parameters $\mathbf{p}^{(l),k}$. The temperature term τ adjusts the bias-variance tradeoff of gradient approximation; as the value of τ approaches 0, samples from the GSM distribution become one-hot. while the variance of the gradients increases. We start at a high τ and anneal to a small but non-zero value as in [13].

3 Experiments

We tested *stochastic filter groups* (SFG) on the problem of simultaneous semantic image regression (synthesis) and segmentation on a prostate radiotherapy dataset. In radiotherapy treatment planning, a CT scan is necessary to allow dose propagation whilst an MRI is required for segmenting organs at risk of ionisation. Instead of acquiring both an MRI and a CT, algorithms can be used to synthesise a CT scan (task 1) and segment organs (task 2) given a single input MRI scan. We acquired 15 3D prostate cancer scans with respective CT and T2-weighted MRI scans with semantic 3D labels for organs (prostate, bladder, rectum and left/right femur heads) obtained from a trained radiologist. We created a training set of 10 patients, with the remaining 5 used for testing. We trained our networks on 2D slices; reconstructing the 3D volumes through patch aggregation.

Baselines: We compared our model against four baselines. They are: 1) single-task networks, 2) hard-parameter sharing multi-task network (MT-hard sharing), 3) SFG-networks with constant $1/3$ allocated grouping (MT-constant mask) as *per* Fig. 1-Right, and 4) SFG-networks with constant grouping probabilities (MT-constant **p**). We note that all four baselines can be considered special cases of the proposed SFG-network: single task networks have SFG shared grouping probability of kernels set to zero; *hard-parameter sharing networks* exists when all shared grouping probabilities are set to 'shared' up until the task-specific layers; and *MT-constant **p*** represents the situation where the grouping is non-informative and each kernel has equal probability of being specific or shared with probability $\mathbf{p}^{(l),k} = [1/3, 1/3, 1/3]$. We used HighResNet [15] as the baseline for CT synthesis and organ segmentation. In our model, we replace each convolutional layer with an SFG module. After the first SFG layer, three distinct repeated residual blocks are applied to $F_{reg}^{(l=0)}$, $F_{seg}^{(l=0)}$, $F_{s}^{(l=0)}$. These are then merged according to the feature routing methodology followed by a new SFG-layer and subsequent residual layers. Our model concludes with 2 successive SFG-layers followed by 1x1 convolutional layers applied to the merged features $F_{reg}^{(l=L)}$ and $F_{seg}^{(l=L)}$. Additional information on training details and dynamics can be found in the supplementary.

Results on CT synthesis and organ segmentation are detailed in Tab. 1. Our method performed best overall in organ segmentation. Our method also obtained best synthesis performance across most anatomical regions; especially in the bone regions when compared against all the baselines. The bone voxel intensities are the most difficult to synthesise from an input MR scan as task uncertainty in the MR to CT mapping at the bone is often highest [3]. Our model was able to disentangle features specific to the bone intensity mapping (Fig. 3-Right) without supervision of the pelvic location, which allowed it to learn a more accurate mapping of an intrinsically difficult task.

3.1 Learned architectures

Analysis of the grouping probabilities allows visualisation of network connectivity and the learned MTL architecture. To analyse the group allocation of kernels at each layer, we computed the sum of class-wise probabilities per layer. Learned grouping allocations are presented in presented in Fig. 3-Left. This illustrates increasing task specialisation in the kernels with network depth. At the first layer, all kernels are classified as

(a) CT Synthesis (PSNR)

Method	Overall	Bones	Organs	Prostate	Bladder	Rectum
One-task (HighResNet) [15]	25.76 (0.80)	30.35 (0.58)	38.04 (0.94)	51.38 (0.79)	33.34 (0.83)	34.19 (0.31)
MT-hard sharing	26.31 (0.76)	31.25 (0.61)	39.19 (0.98)	52.93 (0.95)	34.12 (0.82)	34.15 (0.30)
MT-constant mask	24.43(0.57)	29.10(0.46)	37.24(0.86)	50.48(0.73)	32.29(1.01)	33.44(2.88)
MT-constant **p**=[$^1/_3$,$^1/_3$,$^1/_3$]	26.64(0.54)	31.05 (0.55)	39.11 (1.00)	**53.20 (0.86)**	34.34 (1.35)	35.61 (0.35)
MT-SFG (ours)	**27.74 (0.96)**	**32.29 (0.59)**	**39.93 (1.09)**	53.01 (1.06)	**35.65 (0.44)**	**35.65 (0.37)**

(b) Segmentation (DICE)

Method	Overall	Left Femur Head	Right Femur Head	Prostate	Bladder	Rectum
One-task (HighResNet) [15]	0.848(0.024)	0.931 (0.012)	**0.917 (0.013)**	0.913 (0.013)	0.739 (0.060)	0.741 (0.011)
MT-hard sharing	0.829(0.023)	**0.933 (0.009)**	0.889 (0.044)	0.904 (0.016)	0.685 (0.036)	0.732 (0.014)
MT-constant mask	0.774(0.065)	0.908 (0.012)	0.911 (0.015)	0.806 (0.0541)	0.583 (0.178)	0.662 (0.019)
MT-constant **p**=[$^1/_3$,$^1/_3$,$^1/_3$]	0.752(0.056)	0.917 (0.004)	0.917 (0.01)	0.729 (0.086)	0.560 (0.180)	0.639 (0.012)
MT-SFG (ours)	**0.852 (0.047)**	0.935 (0.007)	0.912 (0.013)	**0.923 (0.016)**	**0.750 (0.062)**	**0.758 (0.011)**

Table 1: Model performance with best results in bold blue, and the second best results in red. Standard deviations are computed over the test subject cohort and shown in brackets.

shared (**p**$= [0, 1, 0]$) as low-order features such as edge or contrast descriptors are generally learned earlier layers. In deeper layers, higher-order representations are learned, which describe various salient features specific to the tasks. This coincides with our network allocating kernels as task specific, as illustrated in Fig. 3. Notably, the learned connectivity of both models shows striking similarities to hard-parameter sharing architectures commonly used in MTL, where there is a set of shared layers aiming to learn a feature set common to both tasks. Task-specific branches then learn a mapping from this feature space for task-specific predictions. Our model learns this structure whilst allowing asymmetric allocation of task-specific kernels with no priors on network structure.

4 Discussion

We have proposed *stochastic filter groups* (SFGs) to disentangle *task-specific* and *generalist* features. SFGs define the grouping of kernels and the connectivity of features in a CNN. We used variational inference to estimate the distribution over connectivity given training data and sample over possible architectures during training. Our method can be considered as a probabilistic form of multi-task architecture search, as the learned posterior embodies the desired MTL architecture given the data.

The concept of disentangling features is important within medical image analysis where the goal is to develop automated tools of clinical utility. There is significant variability in human anatomy whilst many disease phenotypes are prevalent across multiple diseases. Our method offers the possibility to learn shared anatomical and pathological features common across the spectrum of health and disease whilst learning phenotype-

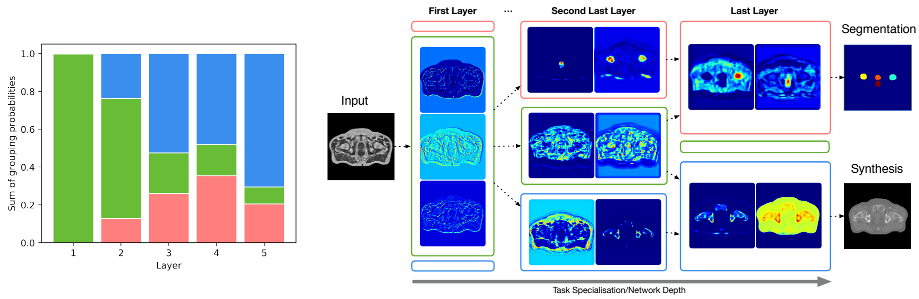

Fig. 3: Left) Learned kernel grouping with ‘CT synthesis’, shared and ‘organ segmentation’ task allocations shown in blue, green and pink; Right) Activation maps from low entropy (high “confidence”) kernels in the learned task-specific and shared filter groups.

specific features. Given a problem where one task consists of tumour segmentation and the second is subtype classification, the shared representation would represent anatomical information important across tasks whilst the subtype latent space may encode information specific across subtypes that can be investigated further for clinical research.

Our method can be exploited for transfer learning. Data scarcity is an issue in medical imaging where labelled data is expensive to acquire. Shared and task-specific representations can be learned on larger datasets and transferred to a new MTL problem with asymmetry in labelled data across tasks. This will be investigated in future work.

Acknowledgements. FB and MJC were supported by CRUK Grant A21993. RT was supported by Microsoft Scholarship. DA was supported by EU Horizon 2020 Research and Innovation Grant 666992, EPSRC Grant M020533, R014019, and R006032, and the NIHR UCLH BRC. We thank NVIDIA Corporation for hardware donation.

References

1. Singla, S., Gong, M., Ravanbakhsh, S., Sciurba, F., Poszos, B., Batmanghelich, K.: Subject2vec: generative-discriminative approach from a set of image patches to a vector. In: MICCAI. (2017)
2. Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M., Thrun, S.: Dermatologist-level classification of skin cancer with deep neural networks. Nature **542** (2017)
3. Bragman, F., Tanno, R., Eaton-Rosen, Z., Li, W., Hawkes, D., Ourselin, S., Alexander, D., McClelland, J., Cardoso, M.J.: Uncertainty in multitask learning: joint representations for probabilistic mr-only radiotherapy planning. In: MICCAI. (2018)
4. Tanno, Ryutaro, M.A., Arslan, S., Oktay, O., Mischkewitz, S., Al-Noor, F., Oppenheimer, J., Mandegaran, R., Kainz, B., Heinrich, M.P.: Autodvt: Joint real-time classification for vein compressibility analysis in deep vein thrombosis ultrasound diagnostics. In: MICCAI. (2018)
5. Caruana, R.: Multitask learning. Machine learning **28**(1) (1997) 41–75

6. Meyerson, E., Miikkulainen, R.: Beyond shared hierarchies: Deep multitask learning through soft layer ordering. In: ICLR. (2018)
7. Misra, I., Shrivastava, A., Gupta, A., Hebert, M.: Cross-stitch Networks for Multi-task Learning. In: CVPR. (2016)
8. Zamir, A.R., Sax, A., Shen, W.B., Guibas, L.J., Malik, J., Savarese, S.: Taskonomy: Disentangling task transfer learning. In: CVPR, IEEE (2018)
9. Ruder, S., Bingel, J., Augenstein, I., Søgaard, A.: Latent multi-task architecture learning. (2019)
10. Lu, Y., Kumar, A., Zhai, S., Cheng, Y., Javidi, T., Feris, R.S.: Fully-adaptive feature sharing in multi-task networks with applications in person attribute classification. In: CVPR. (2017)
11. Mejjati, Y.A., Cosker, D., In Kim, K.: Multi-task learning by maximizing statistical dependence. In: CVPR. (2018)
12. Ioannou, Y., Robertson, D., Cipolla, R., Criminisi, A., et al.: Deep roots: Improving cnn efficiency with hierarchical filter groups. (2017)
13. Gal, Y., Hron, J., Kendall, A.: Concrete dropout. In: NIPS. (2017) 3581–3590
14. Jang, E., Gu, S., Poole, B.: Categorical reparameterization with gumbel-softmax. arXiv preprint arXiv:1611.01144 (2016)
15. Li, W., Wang, G., Fidon, L., Ourselin, S., Cardoso, M.J., Vercauteren, T.: On the compactness, efficiency, and representation of 3d convolutional networks: Brain parcellation as a pretext task. (2017)