CONFIDENTIAL EXTENDED ABSTRACT.
DO NOT DISTRIBUTE ANYWHERE.

単一画像変換ネットワークによる 複数タスクと組み合わせタスクの学習

武田 麻奈[1,a)]　柳井 啓司[1,b)]

概要

近年，画像変換タスクにおいて，Multi Task Learning（MTL）の様々な研究が行われている．一般的な MTL のネットワークは，共有エンコーダとタスク固有のデコーダで構成されており，タスク数が増えるとパラメータ数が大幅に増加する．したがって，タスク固有の部分はできるだけ小さくすることが望ましい．本論文では，ごくわずかなタスク固有部分を持つ単一ネットワークを使用した MTL の方法を提案する．提案手法では，タスク条件信号に基づいてアフィン変換を実行する Feature-wise Linear Modulation（FiLM）を使用する．さらに，異なるタスクの組み合わせを単一ネットワークに学習させることを試みた．いくつかの実験により，提案手法を用いて，異なる画像変換タスクとそれらの組み合わせタスクを学習できることを実証した．さらに，モデルサイズが小さいにも関わらず，様々な画像変換タスクの多くで最新のベースラインよりも優れた性能を示した．

1. はじめに

Deep Convolutional Neural Network (CNN) は，様々な画像変換タスクで大きな成功を収めている．一般的な Single Task Learning (STL) に対して，Multi Task Learning (MTL) では 1 つのネットワークで複数のタスクを実行可能である．しかし，多くの MTL のネットワーク [8] では，タスクごとにデコーダやネットワークの一部を入れ替える必要があるため，タスク数の増加によってネットワークが大きくなるという問題がある．それに対して，共有エンコーダと共有デコーダで構成される単一のネットワークによる MTL は，ネットワークのサイズがタスク数に依存しない利点がある．しかし，一般的にタスクごとに異なるアクティベーションの分布を持っているため，単一ネットワークで複数タスクを学習させる場合，タスクごとに異なる分布に調整する必要がある．このような学習を実現する手法の一つに，Feature-wise Linear Modulation (FiLM) [2], [10] がある．FiLM は，条件信号に基づいたアフィン変換をネットワークに適用することで，タスクごとに異なるアクティベーション分布に調整を行う．

また，Dumoulin らの研究 [3] では，FiLM の特性を生かし，スタイル変換において，複数のスタイルの転送方法を提案した．しかし，複数の異なる画像変換タスクの実行については検証されていない．よって本論文では，単一ネットワークを用いた複数の異なる画像変換タスクの学習と，単一ネットワークによる複数タスクの組み合わせ学習について検討する．

本論文では，組み合わせ学習方法として (1) 順次組み合わせと (2) 領域マスキングによる組み合わせの 2 種類のタスクの組み合わせを想定しており，セクション 3.3 で説明する．

2. 関連研究

Multi-task Learning（MTL）. MTL において，複数のタスクを共同学習することが精度向上につながると示されている．しかし，複数タスクでの特徴共有は，共有する組み合わせによっては悪影響も与えうる（タスク干渉）．本論文では，タスク干渉を緩和させる方法として Feature-wise Linear Modulation (FiLM) [2], [10] を用いる．また，多くの MTL のネットワークはネットワークの一部にタスク固有部分を持っており，タスクの数に比例してパラメータ数が増加するという問題がある [8].

MTL において，異なる画像変換タスクを組み合わせたタスクの学習方法は現在提案されていない．最も類似した研究の一つとして，Sym-parameterized Generative Network(SGN) [1] がある．SGN では，重み付き損失関数を変化させて，複数ドメインの分布を学習し，画像を動的に混合ドメインに変換する． 単一ネットワークを使用し，組み合わせ学習を行う点は本論文と類似しているが，組み合わせるタスクはスタイル変換とドメイン変換であり，類似したタスクに限られる．

Feature-wise Linear Modulation (FiLM). FiLM [2], [10] は，条件付きアフィン変換を形式化したものである．入力に基づいてアフィン変換をネットワークに適用することにより，ネットワークの出力に与える影響を学習する．具体的には，式 (1) のように，条件信号 $\mathbf{c}$ を入力としてスケーリング係数 γ_i およびシフト係数 β_i を出力する関数 f_γ および f_β を学習する．添字 i は特徴または特徴マップの番号を意味する．ここで，式（2）で表されるように，γ_i および β_i は，ネットワークのアクティベーションである $\mathbf{F}_i$ を調整する．

$$\gamma_i = f_{\gamma,i}(\mathbf{c}) \quad , \quad \beta_i = f_{\beta,i}(\mathbf{c}) \tag{1}$$

$$FiLM(\mathbf{F}_i|\gamma_i\beta_i) = \gamma_i\mathbf{F}_i + \beta_i \tag{2}$$

Dumoulin らの研究 [3] では，FiLM をスタイル変換に適用することで，複数スタイルの組み合わせを可能にしている．本論文では，FiLM を用いた，単一ネットワークで複数タスクの学習を行う手法を提案する．提案手法では，Dumoulin らの研究をスタイル変換から複数の画像変換タスクまで拡張させる．

[1] 電気通信大学 大学院情報理工学研究科 情報学専攻
[a)] takeda-m@mm.inf.uec.ac.jp
[b)] yanai@cs.uec.ac.jp

3. 手法

本論文では，単一ネットワークによる複数の異なる画像変換タスクの学習に Feature-wise Linear Modulation (FiLM) [2], [10] を用いる．FiLM は，様々な画像変換タスクにおいてその有効性が示されている．したがって，FiLM は画像変換タスクにおいて高い汎用性を持ち，単一ネットワークで様々な画像変換タスクの学習が可能であると言える．

3.1 タスク指定ベクトル

ネットワークへの，タスクの指定方法としてタスク指定ベクトルを用いる．学習時には，ネットワークに学習させるタスクに応じたタスク指定ベクトル入力し，推論時には，実行したいタスクに応じたタスク指定ベクトルを入力する．タスク数が n の場合，タスク指定ベクトル $\mathbf{c}$ は n 次元のベクトル $[c_1, ..., c_n]$ として定義し，one-hot ベクトルでタスクを指定する．具体的には，タスク数 $n = 3$ でタスク 1 を学習する場合は $\mathbf{c} = [1.0, 0.0, 0.0]$ となる．ここで，$\mathbf{c}$ がゼロベクトル ($c = [0.0, 0.0, ..., 0.0]$) は入力をそのまま出力する恒等変換を表す．学習に恒等変換を含めることで，タスク条件ベクトルに 0.0〜1.0 の中間値を与えることによる，変換の強さの制御が可能になる．

3.2 ネットワークアーキテクチャ

FiLM を使用し，FiLM ジェネレータと FiLM ネットワークからなる図 1 のようなネットワークを構築する．FiLM ネットワークには，Johnson らの研究 [6] の Resblock 付き Encoder-Decoder CNN を用いる．ただし，正規化レイヤとして，Instance Normalization（IN）を使用し，最終レイヤを除くすべてのレイヤの後に FiLM レイヤを挿入する．IN と FiLM の組み合わせは，AdaIN と同じである [5]．FiLM レイヤは，FiLM ジェネレータから FiLM パラメータを受けとり，FiLM パラメータに基づいたアフィン変換により，ネットワークの動作を制御する．

FiLM ジェネレータは，StyleGAN[7] のマッピングネットワークを参考に Fully Connected (FC) 層のみで構築する．FC 層では，FiLM レイヤ番号 i における FiLM パラメータ γ_i，β_i を生成する．FiLM レイヤの計算は，式 (1)，式 (2) に基づき，IN レイヤの後に FiLM ジェネレータによって取得された FiLM パラメータ γ_i，β_i を使用して行う．ここで，γ はスケーリングパラメータ，β はバイアスパラメータを表す．

3.3 組み合わせ学習

提案手法を用いて，複数タスクの学習に加えて組み合わせ学習を行うために (1) 個々のタスクのみ（恒等変換を含む）を学習，(2) 組み合わせるタスクの合計損失関数によって組み合わせタスクを学習，(3) 作成した組み合わせタスクの教師画像で組み合わせタスクを学習する 3 つの方法について検討した．

方法 1 では，単一のタスクのみを個別に学習する．組み合わせタスクの推論は，FiLM ベースの複数スタイルの転送 [3] と同じ方法で実行されると仮定する．方法 2 では，組み合わせる複数タスクの合計損失関数を使用して，組み合わせタスクを学習する．合計損失 L_{mixed} は，重み付けはせずに，タスク A と B の学習結果から得られる L_{taskA} と L_{taskB} を用いて，$L_{mixed} = L_{taskA} + L_{taskB}$ と定義する．方法 2 と方法 3 では，単一タスクのみで構成されていたタスクセットに組み合わせタスクを追加する．組み合わせタスクのタスク指定ベクトルは，タスク数 $n = 3$ でタスク 1 とタスク 3 の組み合わせ学習を行う場合，$c = [1.0, 0.0, 1.0]$ の形式で表される．方法 3 では，組み合わせタスクの教師画像を新しく作成する．図 3 に示すように，組み合わせタスクの教師画像を作成する方法は (a) 順次組み合わせと (b) マスキングによる組み合わせの 2 つがある．順次組み合わせでは，最初に画像修復を適用し，次にスタイル変換を適用するなど，複数の画像変換タスクを順次組み合わせる．一方，マスキングによる組み合わせでは，特定の物体領域のみのスタイル転送など，セマンティックセグメンテーションによって推定された領域マスクを使用して，1 つのタスクによって処理された出力画像をマスクする．ここで，スタイル変換の教師画像は，学習中の同ネットワークでスタイル変換タスクのみを実行した時の出力結果を用いる．組み合わせタスクの学習には，L2 損失を使用する．

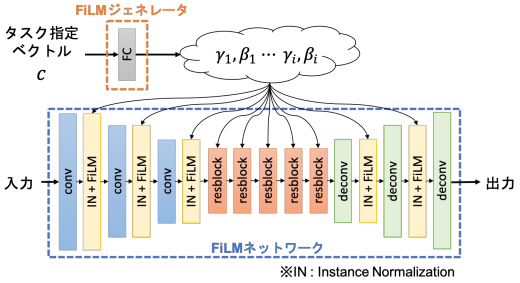


図 1 ネットワークアーキテクチャ．

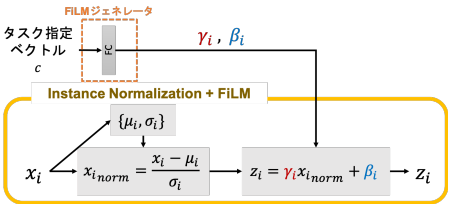
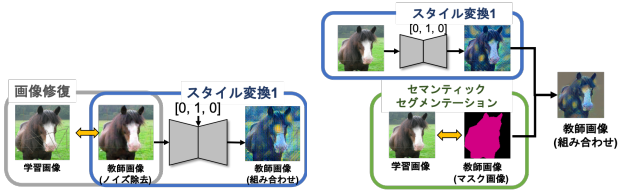


図 2 IN+FiLM レイヤにおける計算．



(a) 順次組み合わせ (b) マスキングによる組み合わせ

図 3 組み合わせ学習における教師画像作成方法．

4. 実験

提案手法の性能を検証するために，(1) 複数の異なるタスクの学習，(2)3 つの組み合わせ学習方法の比較，(3) ベースラインとの比較を行った．データセットには，Pascal VOC を使用した．Pascal VOC は，20 クラスのピクセルレベルの注釈がつけられた人や乗り物， 動物などの一般画像のデータセットである．本実験では，Pascal VOC に Hariharan ら [4] が注釈データを作成した 11,355 枚のデータセットのうち 8,498 枚を学習データ，2,857 枚をテストデータとして使用した．入力画像サイズは，128×128 である．ネットワークには，セクション 3.2 で説明している FiLM ベースのネットワークを使用した．FiLM ジェネレータについては，モデルサイズを小さく保ちつつ，十分な結果が得られた，1 層の FC 層を使用した．

表 1　タスクと損失関数

番号	タスク名	損失関数
タスク 0	再構成 (恒等変換)	L2 損失
タスク 1	画像補完	L2 損失 + 敵対的損失
タスク 2	画像修復	L2 損失
タスク 3	セマンティックセグメンテーション	L2 損失 + 敵対的損失
タスク 4	スタイル変換 1 (Gogh)	知覚的損失 [6]
タスク 5	スタイル変換 2 (Munk)	知覚的損失 [6]

4.1　実験のタスクセット

表 1 は，実験で使用した 6 タスクとその損失関数を示している．全てのタスクの入出力は全て 3 チャネル画像である．タスク 0 は，恒等変換に相当する入力画像の再構成である．タスク 3 では，セマンティックセグメンテーションは通常，クロスエントロピー損失を学習に使用し，クラス数と等しいチャネル数を持つセマンティックセグメンテーションマスクを生成する．ただし本論文ではセマンティックセグメンテーションを他のタスクと組み合わせるために，その出力を，指定されたクラスの領域が切り取られ，他の領域が塗りつぶされる 3 チャネル画像とした．また，損失関数には L2 損失と敵対的損失を使用した．学習に使用される教師画像は，データセットの注釈マスクに基づいて人工的に作成した．本実験では，切り取られるクラスに背景クラスを除く 20 クラスを設定した．

4.2　実験 1：複数の異なるタスクの学習

実験 1 では，複数の異なる画像変換タスクを学習し，提案手法を定性的に評価した．表 1 に示す 6 つのタスクを使用して，ネットワークを学習した．タスク指定ベクトルには，ゼロベクトルを使用するタスク 0 をのぞき，one-hot ベクトルを使用した．図 4 は，6 つのタスクの出力結果である．実験結果から，提案手法による複数の異なる画像変換タスクの学習が可能であると言える．

さらに、推論時にタスク指定ベクトルを変化させて各タスクの強さを制御できることを検証した．図 5 はタスク 0 を除くすべてのタスクについて，タスク指定ベクトルの対応する要素に 0.0〜1.0 の中間値を指定することで，タスクの強さを制御できることを示している．実験結果は，様々な画像変換タスクの目的空間が，再構成タスクと目的タスク間の線形結合空間と見なせることを示している．さらに，中間表現を学習することなく，推論時にこの目的空間内でタスクの強さを制御できることを表している．

4.3　実験 2：3 つの組み合わせ学習方法の比較

実験 2 では，提案手法を使用して，様々な画像変換タスクの組み合わせを学習できるかを検証した．検証では，セクション 3.3 で説明した 3 つの方法を比較した．学習タスクには，表 1 に示されている再構築，画像修復，セマンティックセグメンテーション，およびスタイル変換 1 の 4 つのタスクを使用した．そして，これらタスクの全組み合わせパターンについて実験を行った．方法 3 では，画像修復とスタイル変換のペアに順次組み合わせ（図 3(a)）を使用し，他の 2 つのペアにマスキングによる組み合わせ（図 3(b)）を使用した．

図 6 に組み合わせ学習の出力結果を示す．方法 1 と方法 2 は教師画像通りの出力結果が得られていないのに対して，方法 3 では教師画像とほぼ同じ出力が得られており，組み合わせ学習が成功したことがわかる．以上のことから，作成した組み合わせタスクの教師画像を使用することで，組み合わせ学習が可能であると言える．

さらに，図 7 のように，推論時に条件付きベクトルを変

図 4　提案手法による 6 つのタスクの学習結果.

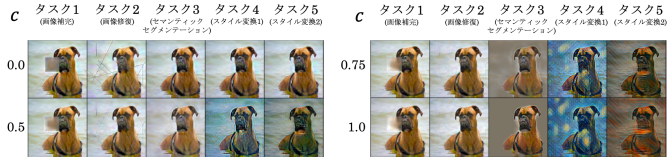


図 5　各タスクの強さの制御. 縦軸は，各タスクに対応するタスク指定ベクトルの要素の値を表す.

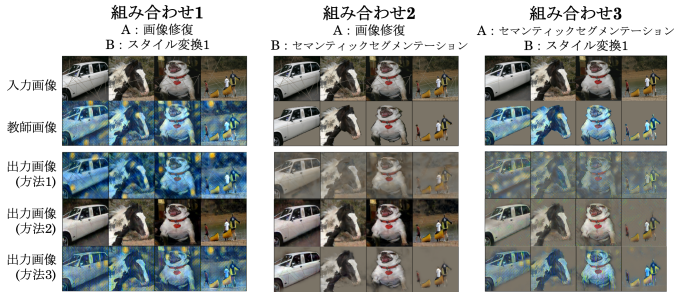


図 6　方法 1，2，3 を用いた組み合わせ学習の出力.

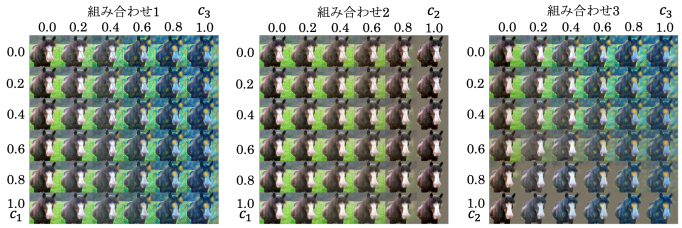


図 7　推論時にタスク指定ベクトルを変化させた時の組み合わせ学習結果（方法 3）.

化させて，恒等変換から組み合わせタスク画像変換への自然な変化が可能であることがわかった．ここで，画像修復，セマンティックセグメンテーション，およびスタイル変換 1 に対応するタスク指定ベクトルの要素は，それぞれ c_1，c_2 および c_3 である．この結果は，提案手法を使用することで，2 つのタスク間の目的空間を 2 つのタスクの線形結合空間とみなすことができ，推論時にこの目的空間で組み合わせタスクの度合いを制御できることを示している．

4.4　実験 3：ベースラインとの比較

実験 3 では，提案手法を定性的および定量的にベースラインと比較する． この実験では，すべてのベースラインのネットワークアーキテクチャは提案手法と同じである．最初のベースラインは，Sym-parameterized Generative Network(SGN) [1] であり，条件付き信号の挿入方法に Conditional Channel Attention Module（CCAM）を採用している．CCAM では，条件付き信号とアクティベーションマップを統合することにより，特徴チャネルを制御する．2 番目のベースラインは，バイアス制御を備えた SGN の CCAM である．3 番目のベースラインは Piggyback [9] である．これは，最初に学習したベースネットワークのパラメータを修正し，新しいタスクが追加されるたびにタスク

表 2　提案手法とベースラインの比較.

		提案手法	EncIN	SGN	SGN+bias	Piggyback1	Piggyback2	Single	SharedEnc
タスク 0	再構成	**8.514**	7.300	3.786	3.777	7.964	7.908	9.129	7.455
	(PSNR⇑, SSIM⇑)	**0.9870**	0.9820	0.9579	0.9578	0.9854	0.9860	0.9892	0.9844
タスク 1	画像補完	**10.306**	9.825	3.445	3.451	9.101	8.139	11.129	6.243
	(PSNR⇑, SSIM⇑))	0.9931	0.9920	0.9553	0.9554	**0.9939**	0.9885	0.9945	0.9806
タスク 2	画像修復	**10.762**	9.838	3.665	3.656	7.590	6.942	10.506	6.839
	(PSNR⇑, SSIM⇑)	**0.9936**	0.9916	0.8703	0.9572	0.9868	0.9830	0.9931	0.9825
タスク 3	セマンティックセグメンテーション	**6.753**	6.386	2.261	2.297	4.369	**6.753**	6.404	5.993
	(PSNR⇑, SSIM⇑, IoU⇑)	0.9798	0.9775	0.9473	0.9500	0.9630	**0.9801**	0.9775	0.9753
		0.5907	0.5689	0.4299	0.4454	0.4030	**0.6014**	0.5639	0.5545
タスク 4	スタイル変換 1 (FID⇓)	**281.3**	318.3	299.0	307.1	333.4.8	331.4	186.7	324.4
タスク 5	スタイル変換 2 (FID⇓)	**235.3**	250.6	263.6	250.1	297.6	323.3	163.9	251.6
組み合わせ 1	画像修復 + スタイル変換 1 (FID⇓)	**304.3**	361.3	349.1	343.4	-	-	-	-
組み合わせ 2	画像修復 +	**6.716**	6.643	2.442	2.470	-	-	-	-
	セマンティックセグメンテーション	**0.9794**	0.9791	0.9524	0.9519	-	-	-	-
	(PSNR⇑, SSIM⇑, IoU⇑)	**0.5755**	0.5752	0.4700	0.4625	-	-	-	-
組み合わせ 3	セマンティックセグメンテーション +	**313.0**	320.3	339.0	348.6	-	-	-	-
	スタイル変換 1 (FID⇓, IoU⇑)	**0.5457**	0.5403	0.5011	0.4555	-	-	-	-
	モデルサイズ (パラメータ数)	1,698,435	1,695,747	1,765,363	1,902,243	**1,688,835**	**1,688,835**	10,075,410	9,572,370

固有のバイナリマスクを学習する方法である．Piggyback1 と Piggyback2 の 2 つのモデルを作成し，それぞれベースネットワークの学習に画像補完とセマンティックセグメンテーションを使用した． Piggyback では組み合わせ学習ができないため，単一タスクのみの結果を比較した．

まず，提案手法とベースラインを定量的に評価した．図 8 に複数の異なる画像変換タスクの学習結果，図 9 に組み合わせ学習の実行結果を示す．組み合わせ学習では，セクション 3.3 で説明されている方法 3 に従い，実験 2 で使用した画像変換タスクに対して学習を行なった．図 8 および図 9 から，提案手法は，特にセマンティックセグメンテーションとスタイル変換のタスクにおいて，定性的にベースラインより優れていたと言える．

さらに，提案手法とベースラインを定量的に評価した．定量的評価には，3 つのベースライン (1)6 つの単一ネットワークのセット (Single)，(2) 共有エンコーダとタスク固有の ResBlocks とデコーダで構成されるネットワーク (SharedEnc)，(3) エンコーダ部分に標準の IN を使用する FiLM ベースのネットワーク（EncIn）を追加した．評価指標には，スタイル変換を含むタスクには rechet Inception Distance（FID），それ以外のタスクには Mean Square Error（MSE）と Structural Similarity（SSIM）を使用した．また，セマンティックセグメンテーションを含むタスクには Intersection over Union（IoU）も使用した．表 2 は，複数の単一タスクと組み合わせタスクを学習した時の，提案手法とベースラインの比較である．表から，提案手法は，モデルサイズが小さいにもかかわらず，ほぼすべてのタスクで最高の評価スコアが得られたことがわかる．

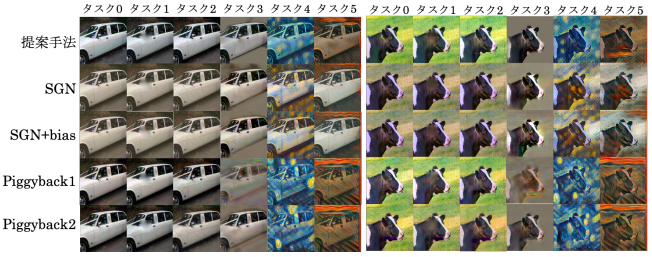


図 8　単一タスクの学習結果の比較出.

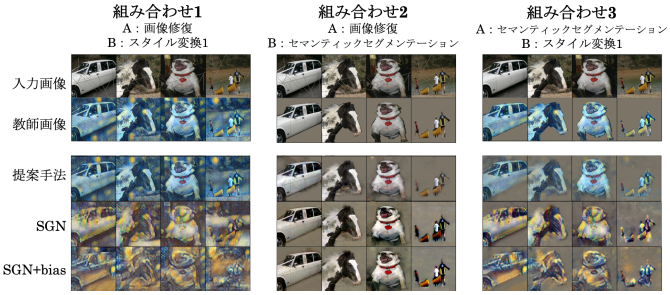


図 9　組み合わせ学習結果の比較.

5. おわりに

本論文では，FiLM ベースの単一ネットワークを使用して，複数の異なる画像変換タスクとそれらの組み合わせ学習を行った．実験から，個々のタスクの学習に加えて，作成した組み合わせタスクの教師画像を使用した組み合わせ学習が可能であることを示した．さらに，各タスクの中間表現と複数のタスク間の相互表現を学習しなくても，推論時にタスク指定ベクトルを変更することにより，これらのタスクの目的空間を補完できることがわかった．また，提案手法を他のベースラインと比較し，その有効性を示した．

参考文献

[1] Chang, S., Park, S., Yang, J. and Kwak, N.: Sym-Parameterized Dynamic Inference for Mixed-Domain Image Translation, *ICCV* (2019).

[2] Dumoulin, V., Perez, E., Schucher, N., Strub, F., Vries, H. d., Courville, A. and Bengio, Y.: Feature-wise transformations, *Distill* (2018). https://distill.pub/2018/feature-wise-transformations.

[3] Dumoulin, V., Shlens, J. and Kudlur, M.: A learned representation for artistic style, *ICLR* (2017).

[4] Hariharan, B., Arbelaez, P., Bourdev, L., Maji, S. and Malik, J.: Semantic Contours from Inverse Detectors, *ICCV* (2011).

[5] Huang, X. and Belongie, S.: Arbitrary Style Transfer in Real-Time With Adaptive Instance Normalization, *ICCV* (2017).

[6] Johnson, J., Alahi, A. and Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution, *ECCV* (2016).

[7] Karras, T., Laine, S. and Aila, T.: A Style-Based Generator Architecture for Generative Adversarial Networks, *CVPR* (2019).

[8] Kokkinos, I.: UberNet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory, *ICCV* (2017).

[9] Mallya, A. and Lazebnik, S.: Piggyback: Adding Multiple Tasks to a Single, Fixed Network by Learning to Mask, *ECCV* (2018).

[10] Perez, E., Strub, F., De Vries, H., Dumoulin, V. and Courville, A.: FiLM:Visual reasoning with a general conditioning layer, *AAAI* (2018).