2. End-to End Multi-Task Learning with Attention

0. Abstract

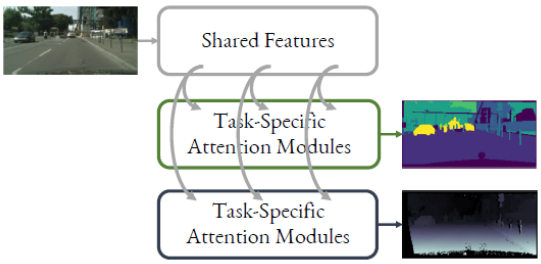
Multi-Task Attention Network(MTAN)

- 전역 feature pool을 포함한 단일 공유 네트워크와 각 작업에 대한 soft-attention 모듈로 구성됨
  - 일반적(global) 특징에서 작업별(task-specific) 특징을 학습할 수 있게 하며 동시에 다른 작업 간에 특징을 공유할 수 있도록 함
- **특징**
  - end-to-end로 훈련할 수 있으며 어떤 feed-forward 신경망(→ 이전 층의 출력이 다음 층으로만 전달되며 순방향으로만 연산이 이루어지는 알고리즘)에도 적용할 수 있음
  - 간단한 구현(매개변수 효율적)
  - 다중 작업 손실 함수의 다양한 가중치 체계에 대해 덜 민감하게 반응함

1. Introduction

1-1. 기존 CNN의 한계

- 기존의 CNN은 이미지 분류, 의미 분할, 스타일 전이를 포함한 여러 컴퓨터 비전 작업에서 큰 성공을 거두었음
  - 그러나 이러한 네트워크들은 일반적으로 특정 작업 하나만 수행하도록 설계됨
  - 메모리 및 추론 속도, 데이터 면에서 각 작업마다 독립적인 네트워크 집합을 구축하는 대신 여러 작업을 동시에 수행할 수 있는 네트워크가 훨씬 더 바람직함
    ⇒ **다중 작업 학습(Multi-Task Learning, MTL)**이 제안됨
- 해당 논문에서는 feature 수준의 주의 마스크(→ 주어진 데이터에서 특성의 중요도를 나타내는 마스크 또는 가중치)를 기반으로 한 MTL 아키텍쳐를 보완



- **공유** 네트워크: 입력 데이터를 받아 작업 공유 특징을 학습
- **주의** 네트워크: 공유 네트워크에 주의 모듈을 적용하여 작업별 특징을 학습

1-2. 기존 MTL의 한계

- 다중 작업 학습(MTL)에는 두 가지 주요 도전 사항이 있음
  1. **네트워크 아키텍처(작업 공유 방법)**
     - 작업 공유(task-shared) 및 작업별 특징(task-specific) 모두를 나타내어야 함
       - task-shared: 네트워크가 일반화된 표현을 학습하도록 장려(overfitting 방지)
       - task-specific: 각 작업에 맞게 특징을 학습할 수 있는 능력 제공(underfitting 방지)
  2. **손실 함수 (작업 균형 조절 방법)**
     - 각 작업의 상대적인 기여도를 가중치로 고려해야 하며, 모든 작업을 동등한 중요성으로 학습할 수 있어야 함

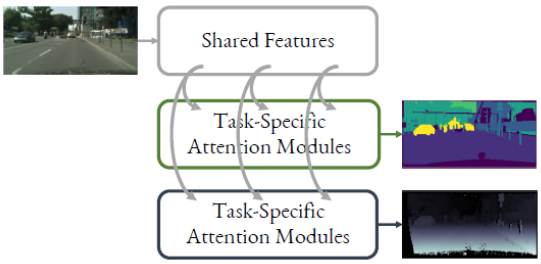
       → 더 쉬운 작업이 우세하게 학습되는 것을 방지해야 함
     - 손실 가중치를 수동으로 조절하는 것은 번거로울 수 있음
       →가중치를 자동으로 학습하거나 다양한 가중치에 강건한 네트워크를 설계하는 것이 선호됨
- 기존의 MTL은 두 가지 도전 과제 중 하나에만 중점을 두며 다른 하나는 표준을 유지하는 것에 초점을 맞추었음

1-3. Multi-Task Attention Network (MTAN)

- 본 논문에서는 위의 두 가지 도전 과제를 통합적으로 해결하는 통합 접근 방식을 소개
  - 자동으로 공통(task-shared) 특징 및 작업별(task-specific) 특징을 모두 학습하도록 설계된 혁신적인 네트워크를 제안
  - 가중치 선택 방법에 대한 내재적인 강건성을 학습
- MTAN은 단일 공유 네트워크로 구성되어 있으며, 모든 작업을 포함하는 특징을 학습하는 전역 특징 풀을 학습함

- 이후 각 작업에 대해 공유 특징 풀에서 직접 학습하는 대신 공유 네트워크의 각 컨볼루션 블록에 소프트 주의(soft-attention) 마스크를 적용

  → 각 주의 마스크가 해당 작업에 대한 공유 특징의 중요성을 자동으로 결정하여 작업 공유 및 작업별 특징을 자기 지도적인 방식으로 학습할 수 있음



- **공유** 네트워크: 입력 데이터를 받아 작업의 공유 특징을 학습
- **주의** 네트워크: 공유 네트워크에 주의 모듈을 적용하여 작업별 특징을 학습

- **기대효과**
  - 작업 간 일반화를 위해 더 풍부한 특징 조합을 학습할 수 있도록 하며 각 개별 작업에 맞게 판별적 특징을 맞춤화할 수 있음
  - 어떤 특징을 공유하고 어떤 것을 작업별로 만들 것인지 자동으로 선택하는 것은 작업을 명시적으로 분리하는 다중 작업 아키텍처에 비해 훨씬 적은 매개변수로 수행 가능(→ 높은 효율성)
  - 작업 유형에 따라 어떤 feed-forward 신경망에든 구축할 수 있음

    ⇒ 여러 기준 모델을 능가하며 다중 작업 학습의 최신 기술과 경쟁력이 있으며, 더 매개변수 효율적이며 따라서 작업 수에 따라 더 원활하게 확장됨
  - 손실 함수에서 가중치 선택에 대한 내구성을 보임

2. 선행 연구(Related Work)

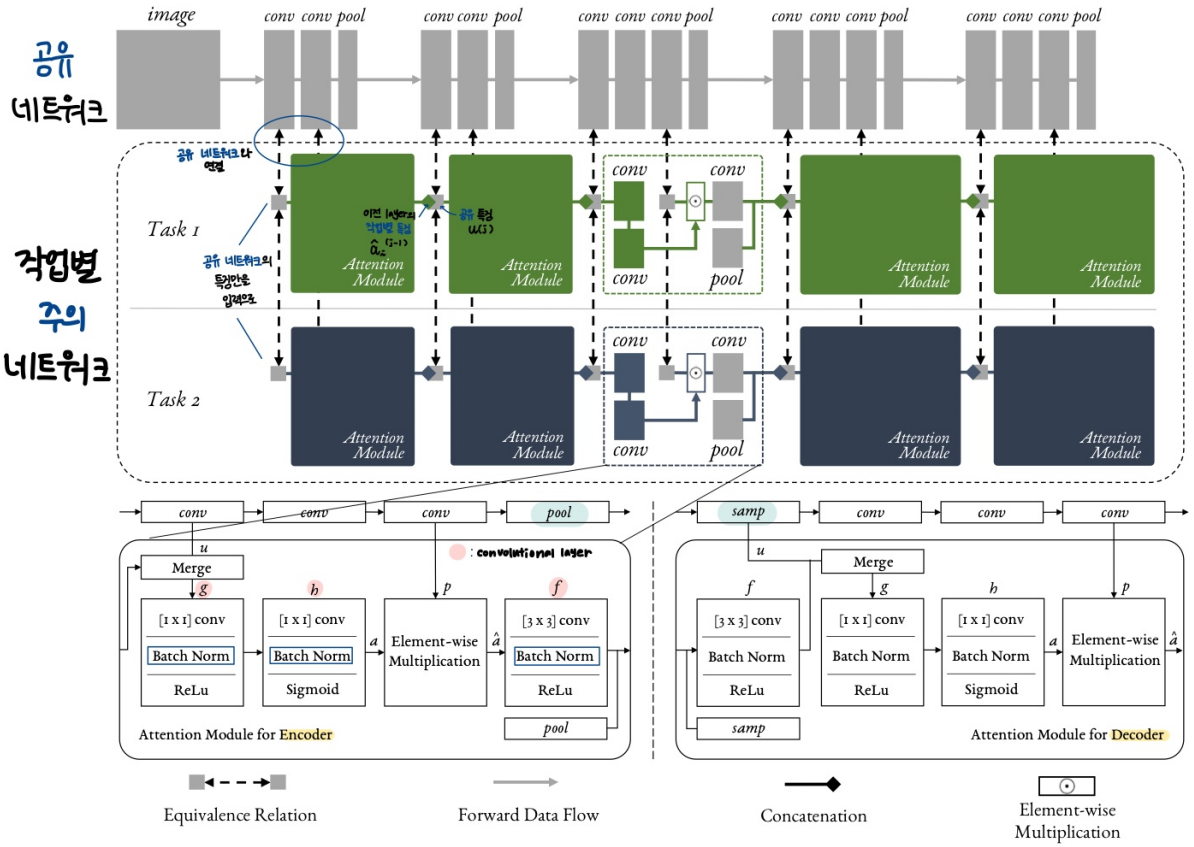
- 다중 작업 학습(Multi-Task Learning, MTL)이라는 용어는 기계 학습에서 널리 사용되었으며, 전이 학습 및 지속적인 학습(→ 모델이 새로운 데이터를 지속적으로 학습하고 적응하면서 이전에 학습한 내용을 잊지 않도록 하는 방법을 연구하는 분야)과 유사성이 있음
- 컴퓨터 비전 분야에서는 여러 도메인에서의 이미지 분류, 포즈 추정 및 동작 인식, 깊이/표면 법선 및 의미 클래스의 밀집 예측과 같은 유사한 작업의 학습에 사용되었음
- 컴퓨터 비전을 위한 대부분의 다중 작업 학습 네트워크 아키텍처는 기존 CNN 아키텍처를 기반으로 설계되었음

    - Cross-Stitch Networks: 각 작업당 하나의 표준 feed-forward 네트워크를 포함하며, 교차 스티치 유닛을 사용하여 특징을 작업 간에 공유할 수 있게 함
    - 자기 지도적 접근법: ResNet101 아키텍처를 기반으로 하며 단일 공유 네트워크의 다른 레이어에서 특징의 정규화된 조합을 학습
    - UberNet: 이미지 피라미드 접근 방식을 제안하여 여러 해상도에서 이미지를 처리하며, 각 해상도에 대해 추가 작업별 레이어가 공유 VGG-Net 위에 형성됨
    - Progressive Networks: 지식을 전송하기 위해 점진적으로 훈련된 네트워크의 일련의 시퀀스를 사용
- **기존 연구의 한계점**
    - Cross-Stitch Networks 및 Progressive Networks와 같은 아키텍처는 많은 수의 네트워크 매개변수가 필요(작업의 수에 선형 비례)
    - 반면, MTAN은 학습 작업당 매개변수가 대략 10% 정도만 증가함
- 다중 작업 학습에서 특징 공유의 균형 맞추기
    - 가중치 불확실성: 작업 불확실성을 사용하여 다중 작업 학습의 손실 함수를 수정
    - GradNorm
        - 훈련 동적을 제어하기 위해 시간에 따른 그래디언트 노름을 조작
        - 작업의 어려움을 결정하기 위해 작업 손실을 사용하는 대신 Dynamic Task Prioritisation은 정확도 및 정밀도와 같은 성능 메트릭을 고려하여 어려운 작업을 직접 우선시하도록 함

3. Multi-Task Attention Network

- **문제 정의**
    - 아키텍처는 어떤 피드포워드 네트워크에든 통합할 수 있지만, 해당 연구에서는 encoder-decoder 네트워크인 `SegNet`을 기반으로 MTAN을 어떻게 구축하는지를 보여줌
    - 의미적 분할(semantic segmentation), 깊이 추정(depth estimation) 및 표면 법선 예측(surface normal prediction)과 같은 이미지 간 밀도 픽셀 수준 예측을 가능하게 함

3-1. 구조(Architect Design)

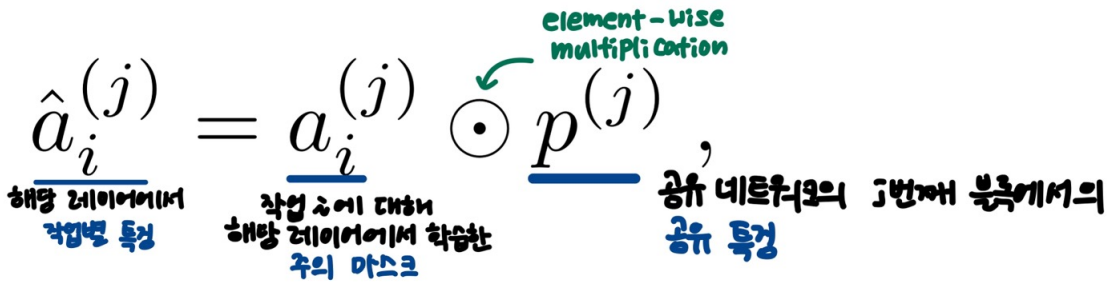


- 하나의 **공유 네트워크**
  - 특정 작업에 기반하여 설계 가능
  - 모든 작업을 대상으로 한 전역 특징 pool을 학습
- K개의 **작업별 주의 네트워크**
  - 공유 네트워크와 연결된 주의(attention) 모듈 집합으로 구성됨
  - 공유 네트워크의 특정 레이어에 소프트 주의 마스크를 적용하여 작업별 특징을 학습
  - 주의 마스크는 공유 네트워크에서의 특징 선택기처럼 작동될 수 있으며, 이들은 자동으로 end-to-end 방식으로 학습
- 주의 마스크는 해당 레이어의 공유 네트워크의 특징에 의존
  - 공유 네트워크의 특징과 소프트 주의 마스크는 함께 학습되어 여러 작업에 걸쳐 공유 특징의 일반화를 극대화하고 동시에 주의 마스크로 인한 작업별 성능을 극대화

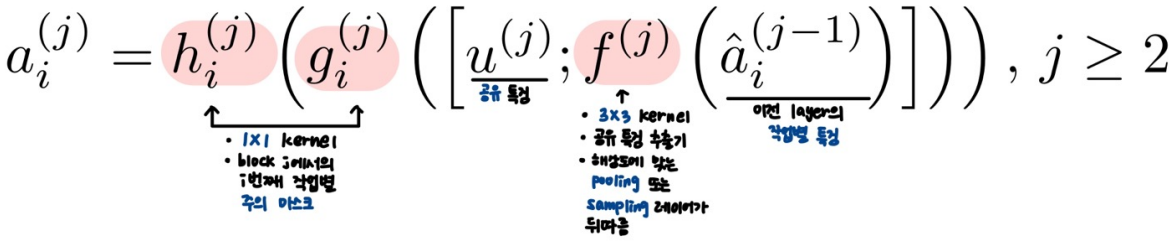
3-2. Task-Specific Attention Module

- 작업별 네트워크가 공유 네트워크의 특징에 소프트 주의 마스크를 적용하여 작업 관련 특징을 학습하도록 설계됨

- 각 작업의 특징 채널당 하나의 주의 마스크가 있음



- encoder의 첫 번째 주의 모듈은 공유 네트워크의 특징만을 입력으로 사용
  - 그러나 블록 j의 후속 주의 모듈의 경우 입력은 공유 특징 u(j)와 이전 레이어의 작업별 특징 ^a_i(j-1)을 연결한 것으로 형성



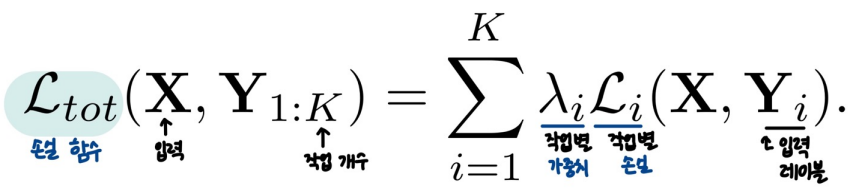
- 주의 마스크는 시그모이드 함수를 거쳐 학습됨
  - $a_i^{(j)}$가 [0, 1] 범위에 있는 것을 보장
  - $a_i^{(j)}$가 1에 가까워지면 마스크가 항등 맵이 되어 주의된 특징 맵은 전역 특징 맵과 동등하게 되며 작업들이 모든 특징을 공유

⇒ 성능이 공유 다중 작업 네트워크의 성능과 비슷하거나 그 이상이 될 것으로 기대

  - 공유 다중 작업 네트워크는 네트워크의 끝에서만 개별 작업으로 분리됨

3-3. The Model Objective

- **손실 함수** 정의



- 이미지 간 **예측** 작업

- 입력 데이터 X에서 레이블 집합 Y_i로의 각 매핑을 평가용으로 하나의 작업으로 간주하며 총 세 가지 작업을 고려
  - **의미적 분할 작업**: depth-softmax 분류기로부터 예측된 각 클래스 레이블에 대해 픽셀 단위로 교차 엔트로피 손실을 적용

$$\mathcal{L}_1(\mathbf{X}, \mathbf{Y}_1) = -\frac{1}{pq}\sum_{p,q}\mathbf{Y}_1(p,q)\log\hat{\mathbf{Y}}_1(p,q).$$

  - **깊이 추정 작업**: 예측된 깊이와 실제 깊이 간의 비교에 L1 Norm을 적용

$$\mathcal{L}_2(\mathbf{X}, \mathbf{Y}_2) = \frac{1}{pq}\sum_{p,q}|\mathbf{Y}_2(p,q) - \hat{\mathbf{Y}}_2(p,q)|.$$

  - 표면 법선 (Surface Normals) 작업(NYUv2에서만 사용 가능): 각 정규화된 픽셀에서 지면 실측 지도와 요소별로 내적을 적용

$$\mathcal{L}_3(\mathbf{X}, \mathbf{Y}_3) = -\frac{1}{pq}\sum_{p,q}\mathbf{Y}_3(p,q)\cdot\hat{\mathbf{Y}}_3(p,q).$$

- 이미지 **분류** 작업
  - 각 데이터셋을 독립적인 도메인의 각각의 개별 분류 작업으로 간주
  - 모든 분류 작업에 대해 표준 교차 엔트로피 손실을 적용

4. Experiments

- 제안된 방법을 두 가지 유형의 작업에서 평가
  - 이미지 간 회귀 작업에 대한 일대다 예측
  - 이미지 분류 작업 (Visual Decathlon Challenge)에 대한 다대다 예측

4-1. 이미지 간 회귀 작업(One-to-Many)

4-1-1. 데이터셋

- **CityScapes**
  - 고해상도 도로 풍경 이미지

        - 두 가지 작업(의미적 분할, 깊이 추정)에 사용
    - 훈련을 가속화하기 위해 모든 훈련 및 검증 이미지 크기를 [128 x 256]로 조정
    - pixelwise 의미적 분할을 위한 19개 클래스와 역 깊이 레이블이 포함되어 있음
    - 깊이 추정 작업은 2, 7 또는 19개 클래스 (7과 19개 클래스에서는 void 그룹을 제외)의 의미적 분할과 함께 진행됨
        - 19개 클래스 및 7개 카테고리에 대한 레이블은 원래 CityScapes 데이터셋에 정의된 것과 동일
    - 이후 배경과 전경 객체만 포함하는 2개 클래스 데이터셋을 추가적으로 생성

2-class	7-class	19-class
background	void	void
	flat	road, sidewalk
	construction	building, wall, fence
	object	pole, traffic light, traffic sign
	nature	vegetation, terrain
	sky	sky
foreground	human	person, rider
	vehicle	carm truck, bus, caravan, trailer, train, motorcycle

        - 7개 클래스 CityScapes 데이터셋에서 다중 작업 학습을 수행

- **NYUv2**
    - RGB-D 실내 장면 이미지
    - 13개 클래스의 의미적 분할, Microsoft Kinect에서 기록된 실제 깊이 데이터, 표면 법선을 평가
    - 훈련을 가속화하기 위해 모든 훈련 및 검증 이미지의 크기를 [288 x 384]로 조정함
    - CityScapes와 비교하면 NYUv2는 실내 장면의 이미지를 포함하고 있으며, 뷰포인트가 크게 다를 수 있고 변화하는 조명 조건이 존재하며, 각 객체 클래스의 모양과 질감이 크게 다를 수 있기 때문에 훨씬 복잡함

4-1-2. Baseline

- MTAN은 일반적이며 어떤 feed-forward 신경망에도 적용할 수 있음

  ⇒ 공정한 비교를 위해 `SegNet`을 기반으로 한 5가지 다른 네트워크 아키텍처 (2개의 단일 작업 + 3개의 다중 작업)를 구현

  1. 단일 작업, 하나의 작업 (Single-Task, One Task): 단일 작업 학습을 위한 기본 SegNet

2. 단일 작업, STAN (Single-Task Attention Network): 단일 작업만 수행하면서 제안된 MTAN을 직접 적용한 단일 작업 주의 네트워크
3. 다중 작업, 분할(Multi-Task, Split - Wide, Deep)
    - 각 특정 작업에 대한 최종 예측을 위해 마지막 레이어에서 분할(Split)하는 표준 다중 작업 학습
    - Split의 두 가지 버전
        - Wide: 컨볼루션 필터 수를 조정함
        - Deep: 컨볼루션 레이어 수를 조정하여 Split이 MTAN과 동일하거나 그 이상의 매개 변수를 가질 때까지 조정함
4. 다중 작업, 밀집(Multi-Task, Dense)
    - 공유 네트워크 및 작업별 네트워크를 포함
    - 각 작업별 네트워크는 주의 모듈 없이 공유 네트워크에서 모든 특징을 받음
5. 다중 작업, 크로스 스티치(Multi-Task, Cross-Stitch): 적응형 다중 작업 학습 접근 방식으로, 이를 SegNet에 구현함

4-1-3. Dynamic Weight Average(DWA)

- 각 작업의 손실 변화율을 고려하여 시간에 따라 작업 가중치를 평균화하는 방법을 학습
- GradNorm에서 영감을 받은 적응형 가중치 방법
    - GradNorm은 네트워크의 내부 그래디언트에 접근해야 함
    - DWA는 숫자적 작업 손실만 필요로 하므로 구현이 훨씬 간단
- DWA에서는 각 작업 k의 가중치 λ_k를 다음과 같이 정의

$$\lambda_k(t) := \frac{K \exp(w_k(t-1)/T)}{\sum_i \exp(w_i(t-1)/T)}, \quad w_k(t-1) = \frac{\mathcal{L}_k(t-1)}{\mathcal{L}_k(t-2)}, \tag{7}$$

softmax operator / 작업 가중치의 부드러움을 제어하는 온도 / 범위 (0, +∞) 내에서의 상대 하강 목표 / 여러 반복에서 각 epoch의 평균 손실

식이 이해되지 않는다..^^;(다시 정리하기)

4-1-4. 결과

- 7-class 버전의 CityScapes 데이터셋과 13-class 버전의 NYUv2 데이터셋을 사용하여 위의 5개의 baseline과 MTAN에 대해 평가

1. **학습(train)**

- 각 네트워크 아키텍처에 대해 세 가지 유형의 가중치 방법을 사용하여 실험 수행
  - 동등한 가중치
  - 가중치 불확실성
  - DWA (하이퍼파라미터: 온도 T = 2, 모든 아키텍처에 대해 경험적으로 최적임을 발견)
- optimizer: Adam(learning_rate: 10^-4)
- batch size
  - CityScapes: 8
  - NYUv2: 2
- iteration: 80K
  - 40K 반복에서 학습률을 반으로 줄임

2. **결과(Results)**

- CityScapes 데이터셋: **MTAN 방법**이 네트워크 매개 변수 수가 절반 미만인 상태에서 Dense 기준선과 유사한 성능을 발휘하며, 다른 모든 baseline model을 능가

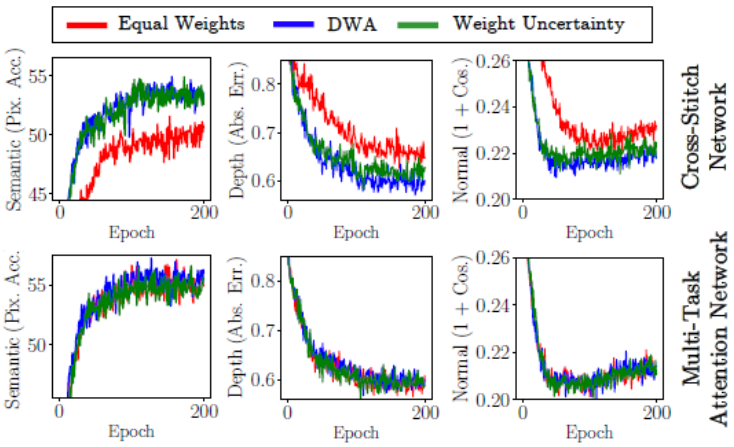
#P.	Architecture	Weighting	Segmentation (Higher Better) mIoU	Segmentation (Higher Better) Pix Acc	Depth (Lower Better) Abs Err	Depth (Lower Better) Rel Err
2	One Task	n.a.	51.09	90.69	0.0158	34.17
3.04	STAN	n.a.	51.90	90.87	0.0145	27.46
1.75	Split, Wide	Equal Weights	50.17	90.63	0.0167	44.73
		Uncert. Weights [14]	**51.21**	**90.72**	**0.0158**	44.01
		DWA, $T = 2$	50.39	90.45	0.0164	**43.93**
2	Split, Deep	Equal Weights	**49.85**	88.69	0.0180	43.86
		Uncert. Weights [14]	48.12	88.68	**0.0169**	**39.73**
		DWA, $T = 2$	49.67	**88.81**	0.0182	46.63
3.63	Dense	Equal Weights	**51.91**	90.89	0.0138	27.21
		Uncert. Weights [14]	51.89	**91.22**	**0.0134**	**25.36**
		DWA, $T = 2$	51.78	90.88	0.0137	26.67
≈2	Cross-Stitch [20]	Equal Weights	50.08	90.33	0.0154	34.49
		Uncert. Weights [14]	50.31	90.43	**0.0152**	**31.36**
		DWA, $T = 2$	**50.33**	**90.55**	0.0153	33.37
1.65	MTAN (Ours)	Equal Weights	53.04	**91.11**	**0.0144**	**33.63**
		Uncert. Weights [14]	**53.86**	91.10	0.0144	35.72
		DWA, $T = 2$	53.29	91.09	0.0144	34.14

- NYUv2 데이터셋: **MTAN 방법**이 모든 가중치 방법과 모든 학습 작업에서 모든 baseline model을 능가

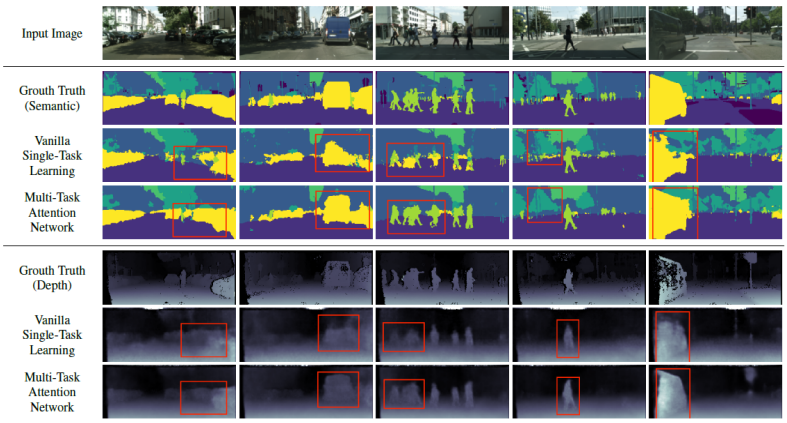
Type	#P.	Architecture	Weighting	Segmentation (Higher Better) mIoU	Segmentation (Higher Better) Pix Acc	Depth (Lower Better) Abs Err	Depth (Lower Better) Rel Err	Surface Normal: Angle Distance (Lower Better) Mean	Angle Distance (Lower Better) Median	Within t° (Higher Better) 11.25	Within t° (Higher Better) 22.5	Within t° (Higher Better) 30
Single Task	3	One Task	n.a.	15.10	51.54	0.7508	0.3266	31.76	25.51	22.12	45.33	57.13
	4.56	STAN	n.a.	15.73	52.89	0.6935	0.2891	32.09	26.32	21.49	44.38	56.51
Multi Task	1.75	Split, Wide	Equal Weights	15.89	51.19	0.6494	0.2804	33.69	28.91	18.54	39.91	52.02
			Uncert. Weights [14]	15.86	51.12	**0.6040**	0.2570	**32.33**	**26.62**	**21.68**	**43.59**	**55.36**
			DWA, $T = 2$	**16.92**	**53.72**	0.6125	**0.2546**	32.34	27.10	20.69	42.73	54.74
	2	Split, Deep	Equal Weights	13.03	41.47	0.7836	0.3326	38.28	36.55	9.50	27.11	39.63
			Uncert. Weights [14]	**14.53**	43.69	0.7705	0.3340	**35.14**	**32.13**	**14.69**	**34.52**	**46.94**
			DWA, $T = 2$	13.63	**44.41**	**0.7581**	**0.3227**	36.41	34.12	12.82	31.12	43.48
	4.95	Dense	Equal Weights	16.06	52.73	0.6488	0.2871	33.58	28.01	20.07	41.50	53.35
			Uncert. Weights [14]	**16.48**	**54.40**	0.6282	0.2761	**31.68**	**25.68**	**21.73**	**44.58**	**56.65**
			DWA, $T = 2$	16.15	54.35	**0.6059**	**0.2593**	32.44	27.40	20.53	42.76	54.27
	≈3	Cross-Stitch [20]	Equal Weights	14.71	50.23	0.6481	0.2871	33.56	28.58	20.08	40.54	51.97
			Uncert. Weights [14]	15.69	52.60	0.6277	0.2702	32.69	27.26	21.63	42.84	54.45
			DWA, $T = 2$	**16.11**	**53.19**	**0.5922**	**0.2611**	**32.34**	**26.91**	**21.81**	**43.14**	**54.92**
	1.77	MTAN (Ours)	Equal Weights	**17.72**	55.32	**0.5906**	0.2577	31.44	**25.37**	**23.17**	45.65	57.48
			Uncert. Weights [14]	17.67	**55.61**	0.5927	0.2592	**31.25**	25.57	22.99	**45.83**	**57.67**
			DWA, $T = 2$	17.15	54.97	0.5956	**0.2569**	31.60	25.46	22.48	44.86	57.24

- **MTAN의 장점**

1. **주의 마스크(**자동으로 어떤 피처를 공유할 것인지 학습**)**와 함께 **단일** 공유 feature pool을 가짐
   - 추가 매개 변수(`column #P`)를 필요로 하지 않으면서도 다른 방법을 능가
   - 경우에 따라서는 훨씬 적은 매개 변수를 가지고 있음
2. 다양한 손실 함수 가중치 방법에 걸쳐 높은 성능을 유지하며, 가중치 방법의 선택에 대한 다른 방법보다 견고함
   - 손실 가중치의 조정을 피할 수 있음



MTAN은 다양한 가중치 방법에 걸쳐 유사한 학습 경향을 따르는 반면, Cross-Stitch Network는 다른 방법에 대해 현저히 다른 동작을 보임

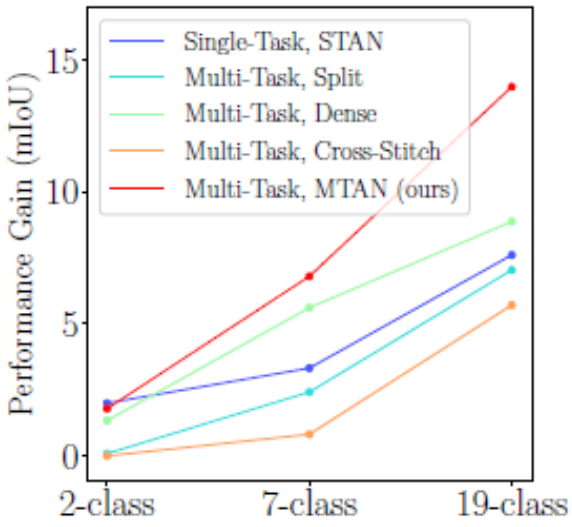


- CityScapes 검증 데이터셋에 대한 질적 결과

- 개체의 가장자리가 더 두드러지는 것으로 보아 MTAN 접근 방식이 일반적인 단일 작업 학습에 비해 이점을 가짐을 확인할 수 있음

4-1-5. 작업 복잡도의 효과

- 다중 작업 학습의 이점을 더 자세히 살펴보기 위해 CityScapes 데이터셋의 서로 다른 semantic 클래스 수를 고려하여 평가
  - 모든 실험에서 깊이 라벨은 동일
  - 모든 네트워크는 동등한 가중치로 훈련시킴



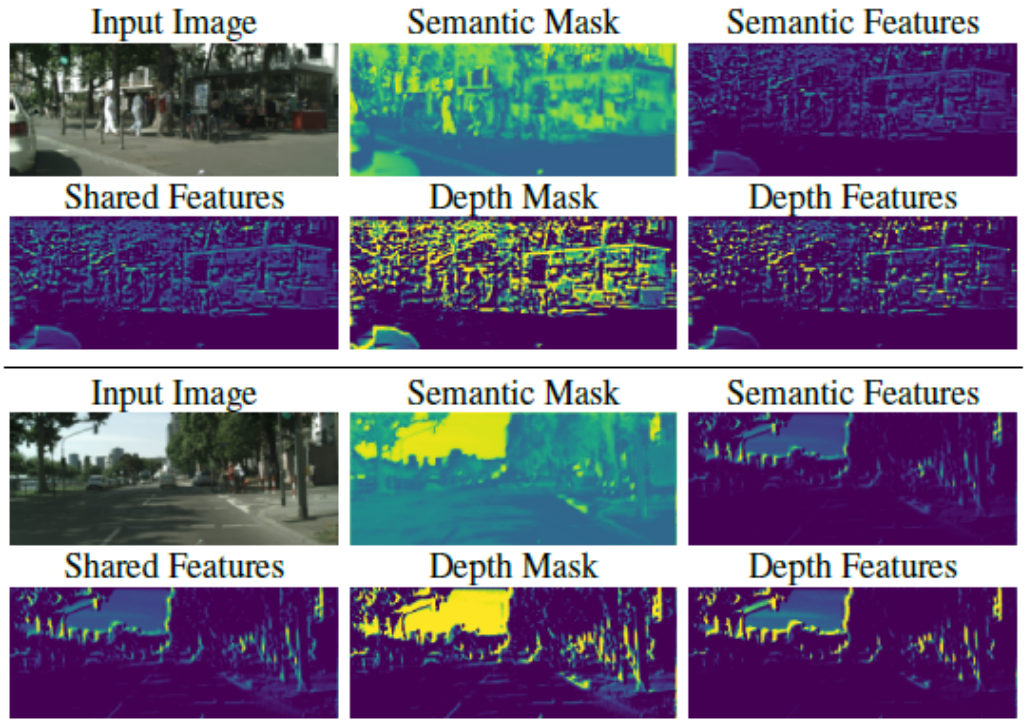
▲ 단일 작업 STAN 구현과 비교한 검증 성능 향상 정도 측정

- 2 클래스 설정에서는 단일 작업 주의 네트워크 (STAN)가 모든 다중 작업 방법보다 더 나은 성능을 발휘
  - 간단한 작업을 위해 네트워크 매개 변수를 완전히 활용할 수 있기 때문에

- 그러나 더 높은 작업 복잡성의 경우, 다중 작업 방법은 가능한 네트워크 매개 변수를 더 효율적으로 활용하기 위해 피처 공유를 촉진하며 더 나은 결과로 이어짐

- 또한 작업 복잡성이 증가함에 따라 모든 구현에 대한 상대적인 성능 향상이 증가하지만, **MTAN** 방법은 더 빠른 속도로 증가함

4-1-6. 특징 선택기로서의 주의 마스크(attention mask)



- 각 마스크는 공유된 피처의 정보가 부족한 부분을 가리고 각 작업에 유용한 부분에 집중하도록 작동함
  - 깊이 마스크는 시맨틱 마스크보다 훨씬 더 높은 대조를 가짐
  - 공유된 피처(shared feature): 시맨틱 작업에 유용
  - 깊이 작업(depth feature): 작업별 피처를 추출하는 데 유용

4-2. 이미지 분류 작업 (Many-to-Many)

- 10가지 개별 이미지 분류 작업 (다대다 예측)으로 구성
- 평가: 작업별 정확도를 보고 이러한 정확도를 기반으로 최대 10,000 (작업당 1,000)의 누적 점수 할당

Method	#P.	ImNet.	Airc.	C100	DPed	DTD	GTSR	Flwr	Oglt	SVHN	UCF	Mean	Score
Scratch [23]	10	59.87	57.10	75.73	91.20	37.77	96.55	56.3	88.74	96.63	43.27	70.32	1625
Finetune [23]	10	59.87	60.34	82.12	92.82	55.53	97.53	81.41	87.69	96.55	51.20	76.51	2500
Feature [23]	1	59.67	23.31	63.11	80.33	45.37	68.16	73.69	58.79	43.54	26.8	54.28	544
Res. Adapt.[23]	2	59.67	56.68	81.20	93.88	50.85	97.05	66.24	89.62	96.13	47.45	73.88	2118
DAN [25]	2.17	57.74	64.12	80.07	91.30	56.54	98.46	86.05	89.67	96.77	49.38	77.01	2851
Piggyback [19]	1.28	57.69	65.29	79.87	96.99	57.45	97.27	79.09	87.63	97.24	47.48	76.60	2838
Parallel SVD [24]	1.5	60.32	66.04	81.86	94.23	57.82	99.24	85.74	89.25	96.62	52.50	78.36	3398
MTAN (Ours)	1.74	63.90	61.81	81.59	91.63	56.44	98.80	81.04	89.83	96.88	50.63	77.25	2941

▲ Visual Decathlon Challenge 온라인 테스트 세트에서의 Top-1 분류 정확도

- 상단 부분: 단일 작업 학습 기준의 결과
- 하단 부분: 다중 작업 학습 기준의 결과

1. **학습**

- Wide Residual Network에 기반한 MTAN을 적용
  - 깊이: 28
  - 너비: 4
  - 각 블록의 첫 번째 합성곱 레이어에서 stride = 2 적용
- 배치 크기: 100
- 옵티마이져: SGD(learning_rate = 0.1, weight_deyac = $5 * 10^-5$)
- epoch: 300
  - 매 50 에포크마다 학습률을 절반으로 줄임
  - 그런 다음 ImageNet을 제외한 9 개의 분류 작업을 수렴할 때까지 학습률 0.01로 미세 조정

2. **결과(Results)**

- MTAN이 대부분의 베이스라인을 능가
- 복잡한 정규화 전략을 필요로 하지 않는 DropOut 적용
- 데이터 세트 크기별로 재구성 또는 각 데이터 세트에 맞춤형 weight_decay 적용 등의 고도의 전략이 필요하지 않음

5. 결론

- 이 연구에서는 다중 작업 학습을 위한 새로운 방법인 **Multi-Task Attention Network(MTAN)**을 제안
  - 전역 특징 풀(global feature pool)과 각 작업에 대한 작업별 주의 모듈(task-specific attention modules)로 구성

    ⇒ 자동으로 작업 공유(task-share) 및 작업별 특징(task-feature)을 end-to-end 방식으로 학습할 수 있음
- CityScapes 및 NYUv2 데이터셋에서 다중 밀집 예측 작업(= 회귀) 및 Visual Decathlon Challenge에서 다중 이미지 분류 작업에 대한 실험 결과는 다른 방법들과 비교하여 **MTAN**이 더 우수하거나 경쟁력이 있다는 것을 시사
- 손실 함수에서 사용되는 특정 작업 가중치 체계에 대한 강건성(robustness)을 나타냄
- 매개 변수 효율적
  - 주의 마스크를 통해 가중치를 공유할 수 있음