|  |
| --- |
| *Taskonomy* : *Disentangling Task Transfer Learning* |
| *Zamir et al., 2018, CVPR* |
| *Read : 2020.04.27-2020.05.04* |

|  |
| --- |
| **Topic** |

Transfer learning using the relationship of visual tasks

|  |
| --- |
| **Problems** |

src task와 trg task를 하나씩 정해두고, 둘 만의 transferability를 계산/사용

🡪 전체를 보면 time / cost가 높음, 사용가능한 data가 있어도 이용불가

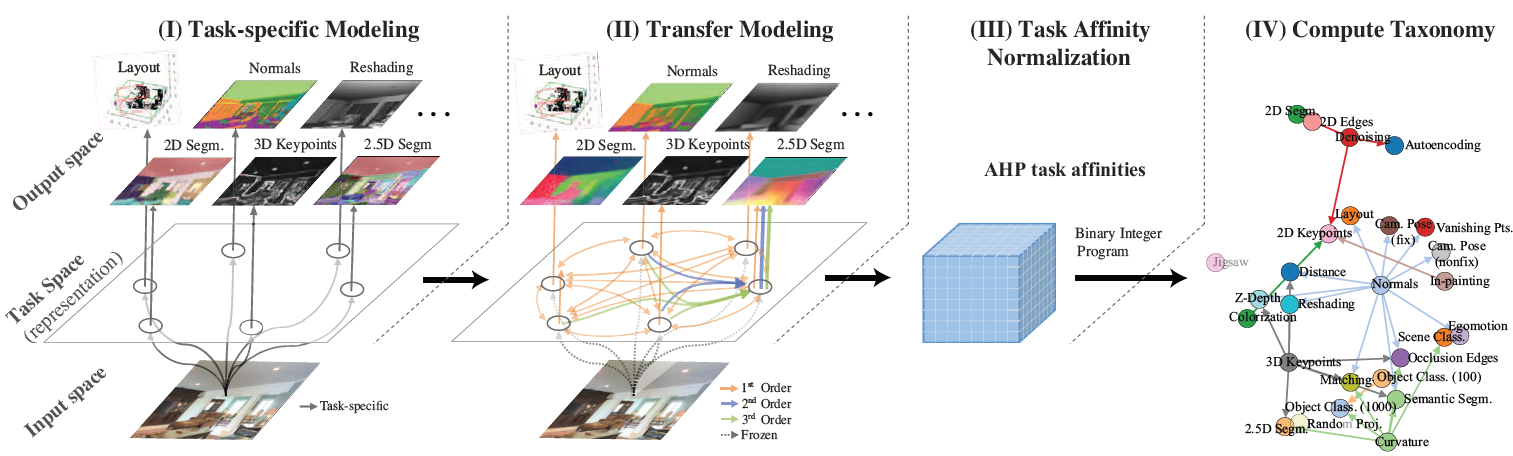
|  |
| --- |
| **Motivation** |

Tasknomy(task taxonomy) : task dictionary(여러 task 조합)에 대해서 각 task 상호 간의 transferability를 위해

🡪 계산적으로 도출가능한 directed hypergraph(task들의 잠재적 관계/구조)

( : 어떤 task를 라벨링하고 어떤 task를 fine-tuning 할지 나타낸 도식)

|  |
| --- |
| **Proposed Method** |



1. **Task-Specific modeling** : src내의 각 task에 대해서 task-특화된 network를 독립적으로 학습
2. **Transfer-Modeling** : src와 trg task 간 조합별 transferability를 수치화 된 형태로 계산

“전체 training set D에 대한 loss를 최소화하는 Θ“

🡪 의 성능 ∝ transferability

🡪 성능 이를 통해 최적의 readout function 도출

: 모든 (src, trg) 조합에 대한 readout function 도출

*: 이미지 I에 대한 transfer network의 예측 label : Decoder ( Encoder (Image) )*

*: 이미지 I에 대한 true label*

*Lt  : Loss Function*

1. **AHP Task Affinities** : 정규화 된 Affinity matrix 도출

k=1, tgt task에 대하여 src task si와 sj를 각각 적용하고 얻은 readout function을 이용하여 test set을 test

test set 중에서 성능이 높은 항목의 개수로 세서, 비율 계산

🡪 이 값으로 pairwise matrix Wt :

🡪 matrix Wt‘ = Wt / WtT: 즉, Si가 Sj에 비해 평균적으로 몇 회나 더 우수하였는지 성능

🡪 Wt‘의 principal eigenvector : 벡터의 i번째 성분은, si­­의 그래프상 centrality (trg에 대한 영향력)

🡪 모든 trg에 대해 Wt‘의 principal eigenvector를 계한사고 row-wise로 쌓으면***🡺 Task Affinity Matrix P***

k>1 , 성능 비교 대상이 src task(1개)가 아니고 – k차 조합 (s1, … , sk , t)

즉, Task Affinity matrix P 중 하나의 k차 조합 (s1, … , sk , t) 중 하나

1. **Compute Taxonomy** : Binary Integer Program(BIP)을 통해, 최적의 Transfer Policy 탐색(:synthesize a hypergraph)

Task Affinity Matrix P에 BIP를 적용

🡪 특정 trg task의 성능을 극대화하는 **최적의 transfer policy**탐색 : **(s1, … , sk , t)** 조합

**BIP** 문제에서 obj function weight 설정 : |E|+|V|개 원소 중,

앞에 |E|개의 transfer edge에 ci :=rtarget(i)〮pi , 뒤 |V|개에는 0입력



(제약) subgraph상에 특정 transfer가 포함되면, 해당 transfer의 src task 또한 반드시 포함 (그래프 모양)

각 trg task로 딱 하나의 transfer가 입력 (하나의 trg task를 위해서는 한 가지의 방법만 존재)

supervision budget을 초과하지 않음

*rtarget(i) : i번째 transfer가 가리키는 trg task에 대한 transferability*

*pi : i번째 transfer가 가리키는 trg tak에 대한 상대적 중요도 - 사용자 입력*

|  |
| --- |
| **Experiments** |

|  |
| --- |
| **Opinion… ? / Proposed** |

|  |
| --- |
|  |

|  |
| --- |
| **Reference** |