Robust Learning Through Cross-Task Consistency

Amir R. Zamir, Alexander Sax, Nikhil Cheerla, Rohan Suri, Zhangjie Cao, Jitendra Malik, Leonidas J. Guibas, CVPR 2020

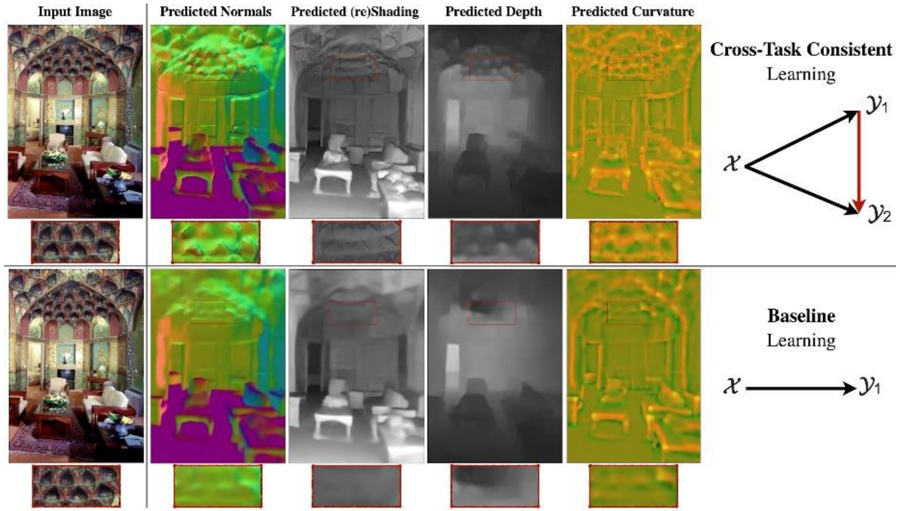
Presented by Miquel Martí

Outline

- Introduction
- Related Work
- Method
  - From elementary consistency unit...
  - ... to globally consistent graphs
- Experiments
  - Accuracy (in/out distribution)
  - Consistency energy utilities
- Conclusion

Introduction

- Visual perception → multiple tasks
  - Predictions from same input are not independent, expected to be consistent (not contradicting)
- **Idea: augment learning with cross-task consistency constraints**
  - Through inference path invariance on a graph of arbitrary tasks, data-driven
  - Better accuracy, better generalization to out-of-distribution samples
- Consistency energy
  - Confidence metric
  - Detect out-of-distribution inputs



Related work

Utilizing consistency

- Consistency constraints proven useful in:
  - Back-translation in NMT and cycle-consistency in image translation
  - Temporal consistency and 3D geometry constraints in vision

Uncertainty metrics

- Consistency energy similar to ensemble averaging
  - Estimations from different paths instead of from different trainings of same model with different initializations / seeds.
  - More effective at capturing uncertainty (?)

Multi-task and transfer learning

- Multiple output domains from a single input, using shared representation

or

- Predict a target output with another task solution as a source

Output domains not automatically consistent, no mechanisms to enforce so

Method

Basic consistency unit

x - input domain

$y = \{y_1, \dots, y_n\}$ – set of output domains

$(x, y_1, \dots, y_n)$ – sample

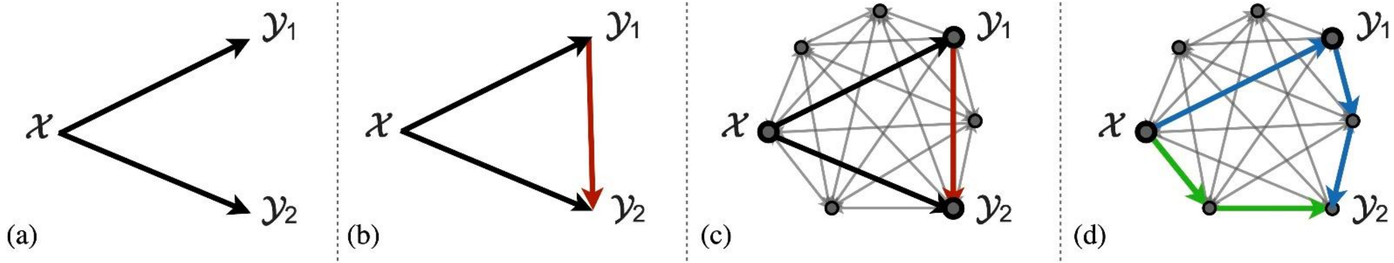
We want to learn functions $f_{x \to y_j}\ \forall\ y_j \in y$ mapping input to output domains

We can however also define functions $f_{y_i \to y_j}\ \forall\ y_i, y_j \in y, i \neq j$ that map output domains, i.e. cross-task functions.

To enforce cross-task consistency between two tasks, use the following loss

$$\mathcal{L}_{xy_1y_2}^{triangle} = |f_{xy_1}(x) - y_1| + |f_{xy_2}(x) - y_2| + |f_{y_1y_2}(f_{xy_1}(x)) - f_{xy_2}(x)|$$

Assume functions $f_{y_i \to y_j}$ are known…



Method

Separability

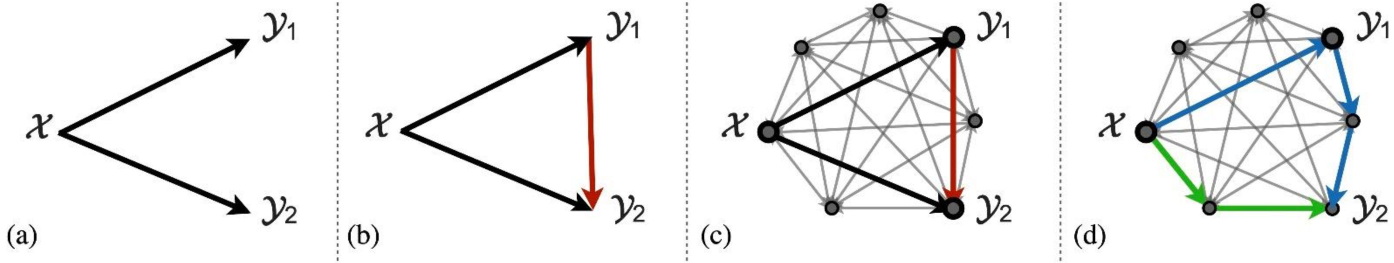
- $\mathcal{L}^{triangle}$ requires simultaneous training of two networks
- We can get a loss separable into functions of f_{xy_1} or of f_{xy_2}

$$\mathcal{L}^{separable}_{xy_1y_2} = \left|f_{xy_1}(\mathrm{x}) - \mathrm{y}_1\right| + \left|f_{y_1y_2}(f_{xy_1}(\mathrm{x})) - \mathrm{y}_2\right|$$

Perceptual loss

- We relied on:
  - perfect mappings $\mathrm{f}_{y_i \to y_j}$
  - datasets with multidomain annotations *for all samples*
- We can get a *perceptual* loss

$$\mathcal{L}^{perceptual}_{xy_1y_2} = \left|f_{xy_1}(\mathrm{x}) - \mathrm{y}_1\right| + \left|f_{y_1y_2}(f_{xy_1}(\mathrm{x})) - f_{y_1y_2}(\mathrm{y}_1)\right|$$

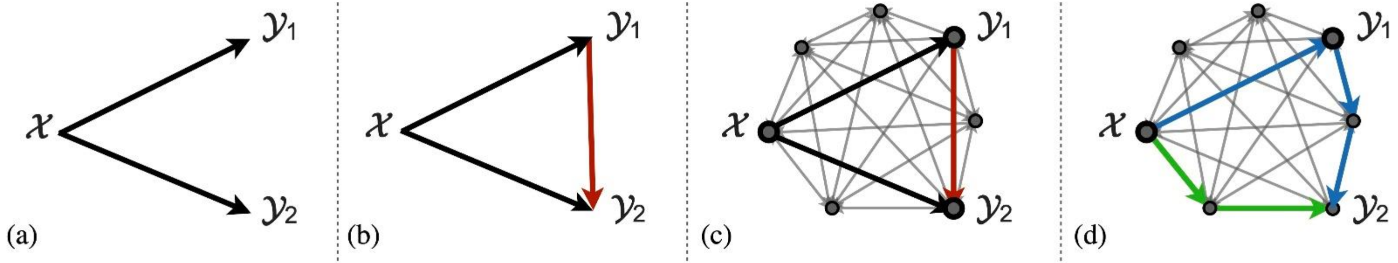


Method

Consistency with multiple domains

Extension to multiple domains is straightforward

$$\mathcal{L}_{xy_1Y}^{perceptual} = |Y| \left| f_{xy_1}(\mathrm{x}) - \mathrm{y}_1 \right| + \sum_{y_i \in Y} \left| f_{y_1 y_i}(f_{xy_1}(\mathrm{x})) - f_{y_1 y_i}(\mathrm{y}_1) \right|$$

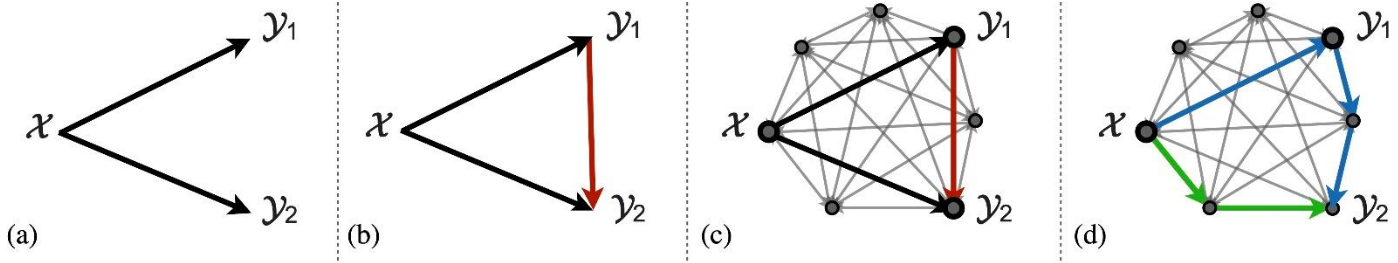


Method

Globally consistent graphs

The objective is formulated over a graph $\mathcal{G} = (\mathcal{D}, \mathcal{F})$ with nodes $\mathcal{D}$ representing all (input and output domains) and edges being the functions mapping between domains.

Now, more generally, we can define the consistency constraints as arbitrary path invariance, i.e. **two paths with the same endpoints should have the same result**.



Method

Globally consistent graphs

Reaching *global* cross-task consistency for graph $\mathcal{G}$ is defined as satisfying the constraint for all feasible paths ($\mathcal{P}$) in $\mathcal{G}$.

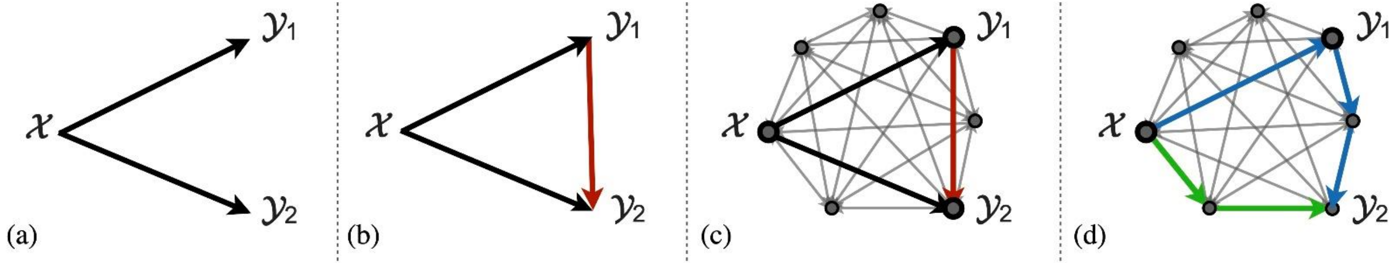
$$\mathcal{L}_{\mathcal{G}} = \sum_{p \in \mathcal{P}} \mathcal{L}_{p}^{perceptual}$$

This is intractable, but an approximate solution similar to *approximate message passing* in graphical models is used.

Algorithm 1: Globally Cross-Task Consistent Learning of Networks $\mathcal{F}$

Result: Trained edges $\mathcal{F}$ of graph $\mathcal{G}$

Train each $f \in \mathcal{F}$ independently. ▷ initialization by standard direct training.
for $k \leftarrow 2$ *to* L **do**
  while *LossConvergence*$(\mathcal{F})$ not met **do**
    $f_{ij} \leftarrow$ *SelectNetwork*$(\mathcal{F})$ ▷ selects target network to be trained.
    $p \leftarrow$ *SelectPath*$(f_{ij}, k, \mathcal{P})$ ▷ selects a feasible consistency path for f_{ij} with maximum length k from $\mathcal{P}$.
    optimize $\mathcal{L}_{ijp}^{perceptual}$ ▷ trains f_{ij} using path constraint p in loss 8.
  end
end



Method

Consistency energy

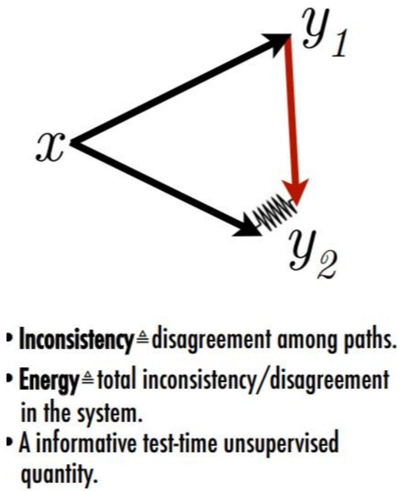
Aim: quantify amount of cross-task consistency.

Standardized average of pairwise inconsistencies:

$$\text{Energy}_{\mathcal{Y}_k}(x) \triangleq \frac{1}{|\mathcal{Y}|-1} \sum_{\mathcal{Y}_i \in \mathcal{Y}, i \neq k} \frac{|f_{\mathcal{Y}_i\mathcal{Y}_k} \circ f_{\mathcal{X}\mathcal{Y}_i}(x) - f_{\mathcal{X}\mathcal{Y}_k}(x)| - \mu_i}{\sigma_i}$$

Turns out to be informative as:

- Confidence/uncertainty metric
- Domain-shift metric



Experiments

Datasets Taskonomy as main training dataset, Replica and NYU for evaluation, CocoDoom and ApolloScape for out-of-training-distribution evaluation

Architecture UNet for all mapping functions

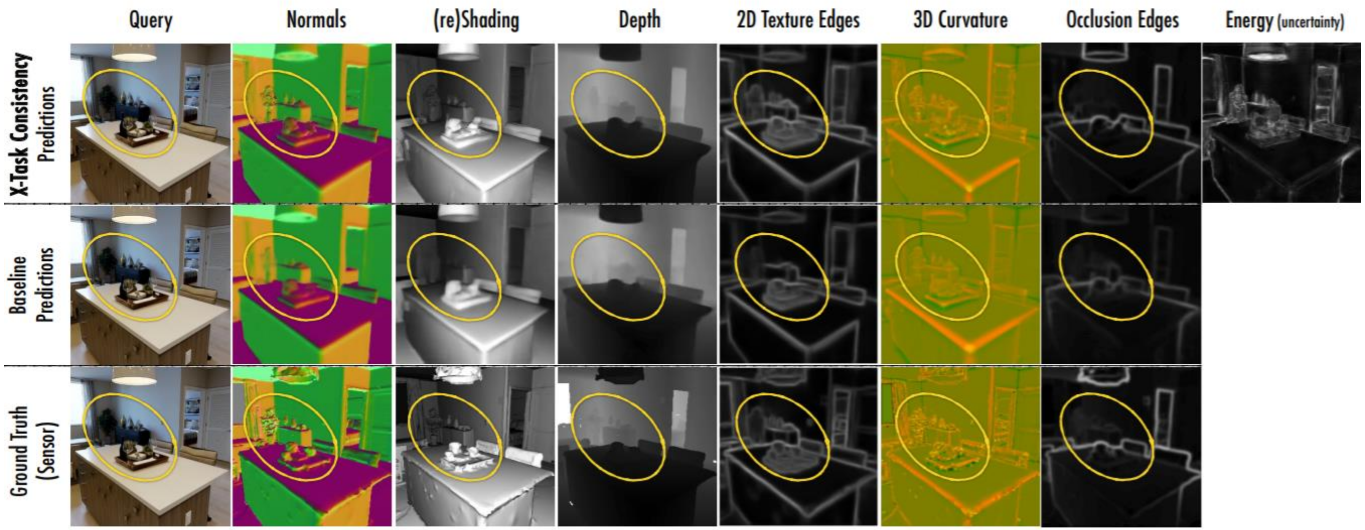
Training details All networks with same details. All losses are L1 and maximum path length in $\mathcal{G}$ set to 3.

Baselines

- Independent learning (main baseline)
- Multi-task learning
- Cycle-consistency - special case triangle consistency with $y_2 = x$
- Baseline perceptual loss - random mappings bw. output domains
- Some others, including curated (not data driven) consistency GeoNet

Experiments

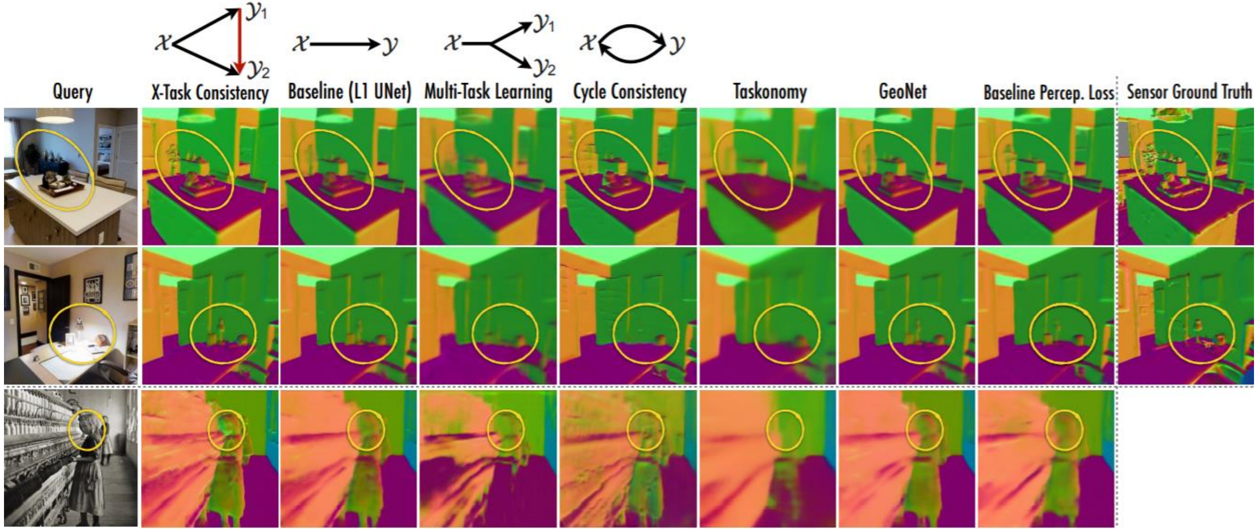
Accuracy of predictions



Experiments

Accuracy of predictions

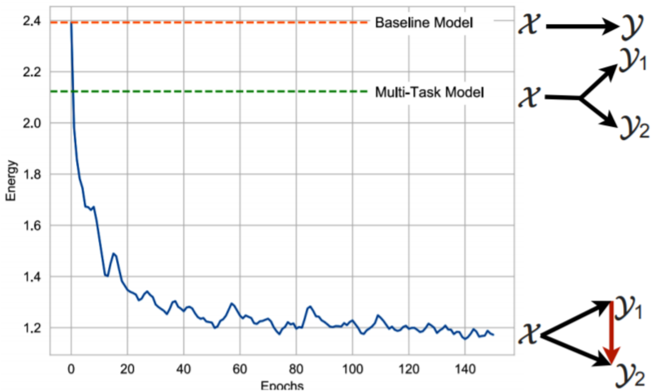
different
input
domain



Experiments

Consistency of predictions

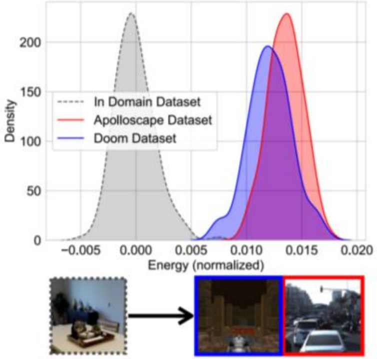
Inconsistency reduced through training with cross-task consistency constraints.
Does not occur with multi-task baseline.



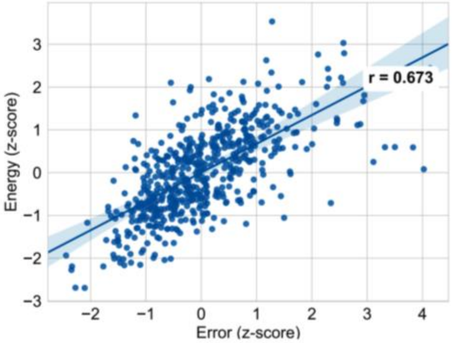
Experiments

Consistency energy utilities

- As a domain shift metric: out-of-distribution data points have high consistency energy. Using this to detect out-of-distribution images yielded ROC-AUC of 0.95.



- As a confidence metric: 0.67 Pearson correlation coefficient with error



Conclusion and limitations

- Cross-task consistency is a **general** and **data-driven** framework to augment standard learning in multiple output domain datasets.
- It gives **more accurate** predictions and **generalizes better** to out-of-distribution data samples.
- **Consistency energy** is informative as **confidence metric** and **domain-shift metric**

- Limitations:
  - Only used in dense prediction tasks, not easily portable to e.g. classification tasks, for which the cross-task functions can be extremely ill-posed.
  - Requires labelled training data in samples with annotations for *some* output domain
  - Added constraints can make the optimizer to have a hard time and/or introduce artifacts