IGN
INSTITUT NATIONAL DE L'INFORMATION GÉOGRAPHIQUE ET FORESTIÈRE
l'information grandeur nature

Superpoint approach to 3D

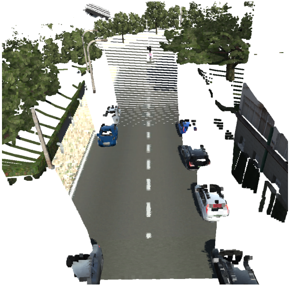
Loic Landrieu

March 2019

Presentation Layout

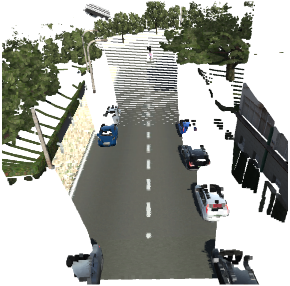
Deep Learnng for 3D data

- **A hard problem**
  - Data volume considerable.



credit: Gaidon2016, Engelmann2017, Hackel2017

Deep Learnng for 3D data

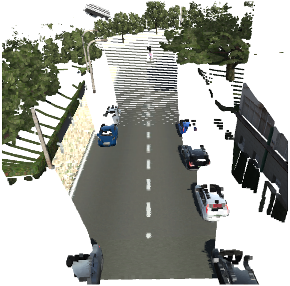


- **A hard problem**
  - Data volume considerable.
  - Lack of grid-structure.

credit: Gaidon2016, Engelmann2017, Hackel2017

Deep Learnng for 3D data

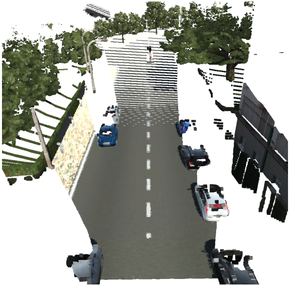
- **A hard problem**
  - Data volume considerable.
  - Lack of grid-structure.
  - Permutation-invariance.



credit: Gaidon2016, Engelmann2017, Hackel2017

Deep Learnng for 3D data

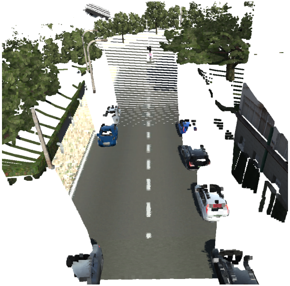
- **A hard problem**
  - Data volume considerable.
  - Lack of grid-structure.
  - Permutation-invariance.
  - Sparsity, variable density.



credit: Gaidon2016, Engelmann2017, Hackel2017

Deep Learnng for 3D data

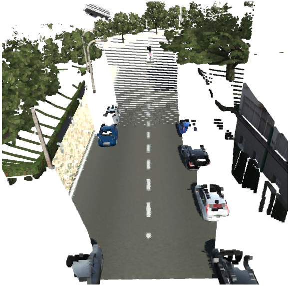
- **A hard problem**
  - Data volume considerable.
  - Lack of grid-structure.
  - Permutation-invariance.
  - Sparsity, variable density.
- What works:
  - set based methods for shape embedding (PointNet)



credit: Gaidon2016, Engelmann2017, Hackel2017

Deep Learnng for 3D data

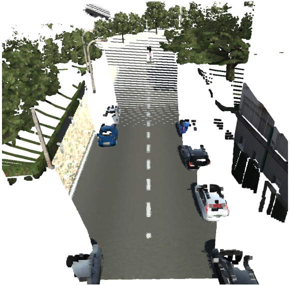
- **A hard problem**
  - Data volume considerable.
  - Lack of grid-structure.
  - Permutation-invariance.
  - Sparsity, variable density.
- What works:
  - set based methods for shape embedding (PointNet)
  - graph convolution for relationships analysis



credit: Gaidon2016, Engelmann2017, Hackel2017

Deep Learnng for 3D data

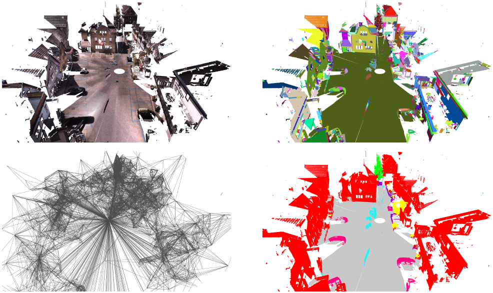
- **A hard problem**
  - Data volume considerable.
  - Lack of grid-structure.
  - Permutation-invariance.
  - Sparsity, variable density.
- What works:
  - set based methods for shape embedding (PointNet)
  - graph convolution for relationships analysis
- **However:** do not scale well at all.



credit: Gaidon2016, Engelmann2017, Hackel2017

SuperPoint-Graph

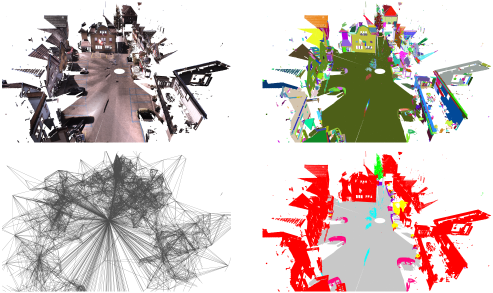
- **Observation:**
  $n_{\text{points}} \gg n_{\text{objects}}$.



Landrieu&Simonovski2018

SuperPoint-Graph

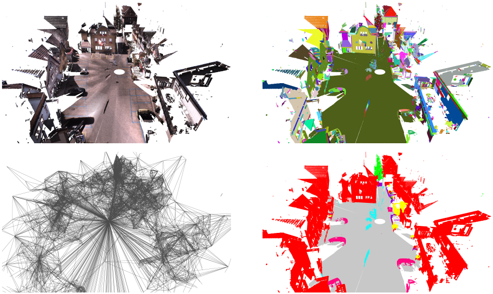
- **Observation:**
  $n_{\text{points}} \gg n_{\text{objects}}$.
- Partition scene into superpoints with simple shapes.



Landrieu&Simonovski2018

SuperPoint-Graph

- **Observation:** $n_{\text{points}} \gg n_{\text{objects}}$.
- Partition scene into superpoints with simple shapes.
- Only a few superpoints, context leveraging with powerful graph methods.



Landrieu&Simonovski2018

Pipeline

- Semantic segmentation down to 3 sub-problems:

Pipeline

- Semantic segmentation down to 3 sub-problems:
- **Geometric Partition :** into simple shapes.

  Complexity: very high (clouds of 10^8 points)

  Algorithm: ℓ_0-cut pursuit

Pipeline

- Semantic segmentation down to 3 sub-problems:
- **Geometric Partition :** into simple shapes.

  Complexity: very high (clouds of 10^8 points)

  Algorithm: ℓ_0-cut pursuit
- **Superpoint embedding:** learning shape descriptors

  Complexity: low (subsampling to 128 points $\times \sim$ 1000 points)

  Algorithm: PointNet

Pipeline

- Semantic segmentation down to 3 sub-problems:
- **Geometric Partition :** into simple shapes.

  Complexity: very high (clouds of 10^8 points)

  Algorithm: ℓ_0-cut pursuit
- **Superpoint embedding:** learning shape descriptors

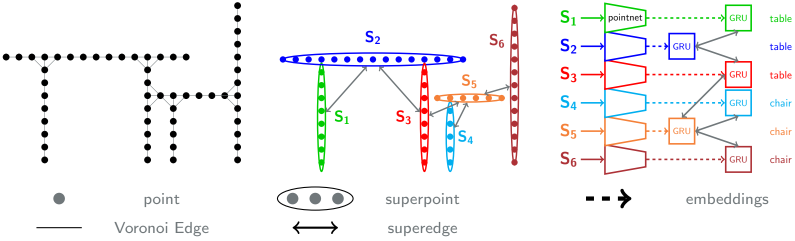
  Complexity: low (subsampling to 128 points $\times \sim$ 1000 points)

  Algorithm: PointNet
- **Contextual Segmentation:** using the global structure

  Complexity: very low (superpoint graph $\sim$ 1000 sp)

  Algorithm: ECC with Gated Recurrent Unit (GRU)

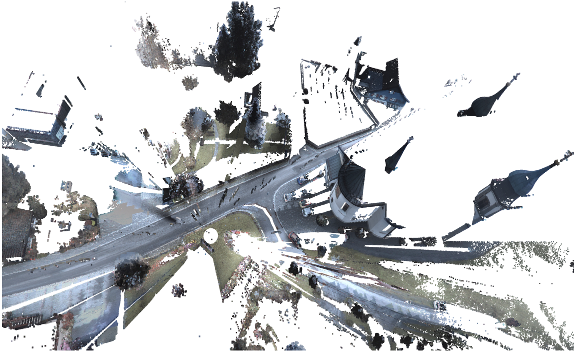
Pipeline



(a) Point cloud (b) Superpoint graph (c) Convolution Network

Qualitative Results: Semantic3D

Semantic3D: 3 billions points over 30 clouds



- route
- herbe
- arbre
- buisson
- bâtiment
- aménagement
- artefact
- voiture

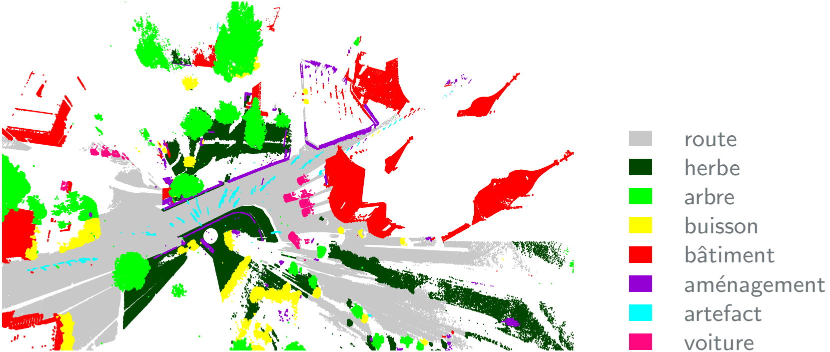
Qualitative Results: Semantic3D

Semantic3D: 3 billions points over 30 clouds

- route
- herbe
- arbre
- buisson
- bâtiment
- aménagement
- artefact
- voiture

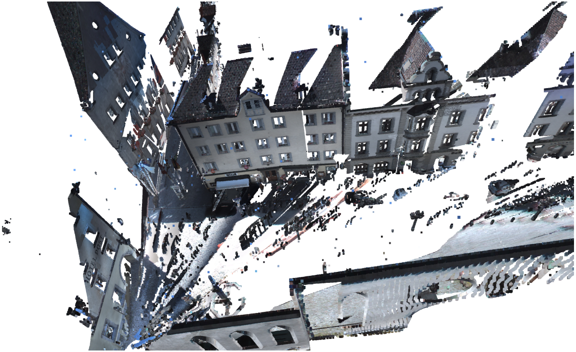
Qualitative Results: Semantic3D

Semantic3D: 3 billions points over 30 clouds



Qualitative Results: Semantic3D

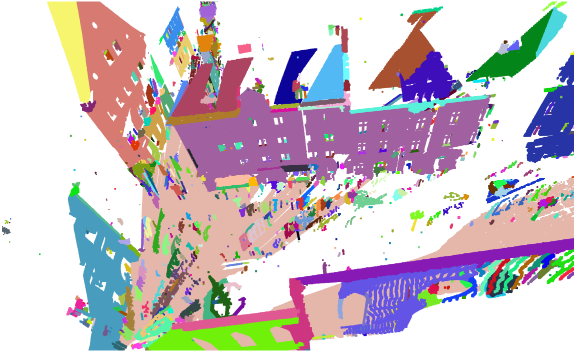
Semantic3D: 3 billions points over 30 clouds



- route
- herbe
- arbre
- buisson
- bâtiment
- aménagement
- artefact
- voiture

Qualitative Results: Semantic3D

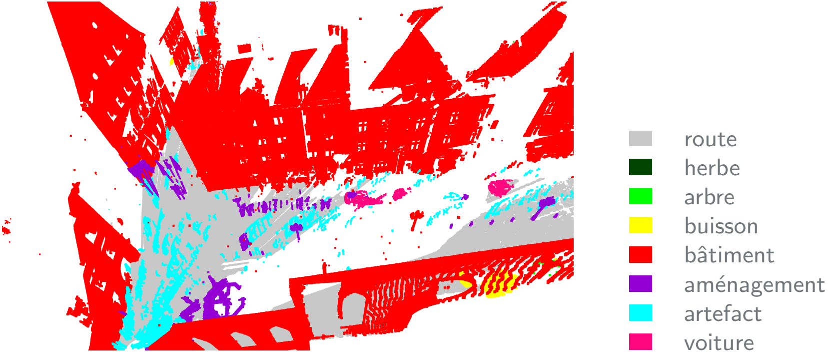
Semantic3D: 3 billions points over 30 clouds



- route
- herbe
- arbre
- buisson
- bâtiment
- aménagement
- artefact
- voiture

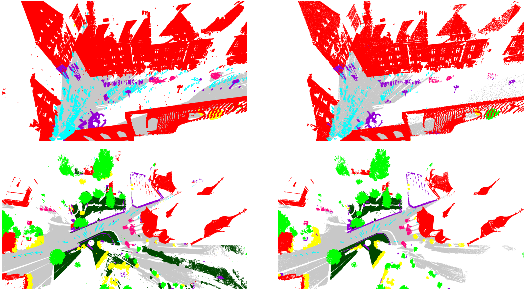
Qualitative Results: Semantic3D

Semantic3D: 3 billions points over 30 clouds



Quantitative Results: Semantic3D

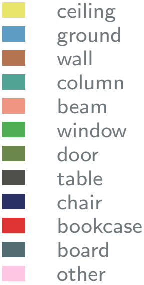
prediction ground truth



Methode	OA	mIoU	road	grass	tree	bush	build-ing	hard-scape	arti-fact	cars
reduced test set: 78 699 329 points										
TMLC-MSR	86.2	54.2	89.8	74.5	53.7	26.8	88.8	18.9	36.4	44.7
DeePr3SS	88.9	58.5	85.6	83.2	74.2	32.4	89.7	18.5	25.1	59.2
SnapNet	88.6	59.1	82.0	77.3	79.7	22.9	91.1	18.4	37.3	64.4
SegCloud	88.1	61.3	83.9	66.0	86.0	40.5	91.1	30.9	27.5	64.3
SPG (Ours)	**94.0**	**73.2**	**97.4**	**92.6**	**87.9**	**44.0**	**93.2**	**31.0**	**63.5**	**76.2**
full test set: 2 091 952 018 points										
TMLC-MS	85.0	49.4	91.1	69.5	32.8	21.6	87.6	25.9	11.3	55.3
SnapNet	91.0	67.4	89.6	**79.5**	74.8	56.1	90.9	36.5	34.3	77.2
SPG (Ours)	**92.9**	**76.2**	**91.5**	75.6	**78.3**	**71.7**	**94.4**	**56.8**	**52.9**	**88.4**

Qualitative Results: S3DIS

Indoor, 3 buildings, 6 stories, 200+ rooms, 600 000 000+ points



Qualitative Results: S3DIS

Indoor, 3 buildings, 6 stories, 200+ rooms, 600 000 000+ points

- ceiling
- ground
- wall
- column
- beam
- window
- door
- table
- chair
- bookcase
- board
- other

Qualitative Results: S3DIS

Indoor, 3 buildings, 6 stories, 200+ rooms, 600 000 000+ points

ceiling
ground
wall
column
beam
window
door
table
chair
bookcase
board
other

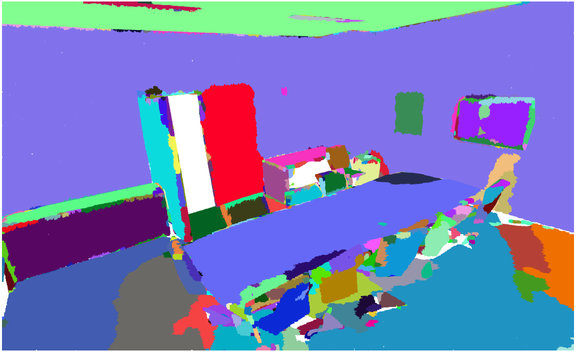
Qualitative Results: S3DIS

Indoor, 3 buildings, 6 stories, 200+ rooms, 600 000 000+ points

- ceiling
- ground
- wall
- column
- beam
- window
- door
- table
- chair
- bookcase
- board
- other

Qualitative Results: S3DIS

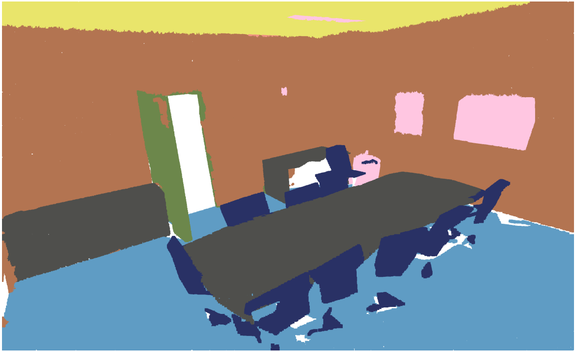
Indoor, 3 buildings, 6 stories, 200+ rooms, 600 000 000+ points

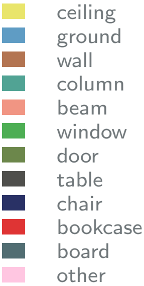


- ceiling
- ground
- wall
- column
- beam
- window
- door
- table
- chair
- bookcase
- board
- other

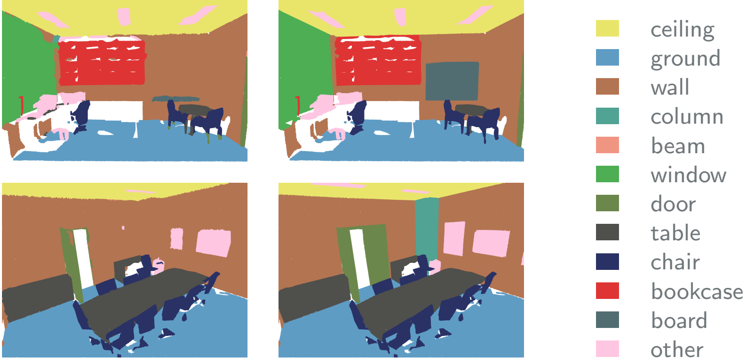
Qualitative Results: S3DIS

Indoor, 3 buildings, 6 stories, 200+ rooms, 600 000 000+ points





Résultats qualitatif: S3DIS



Quantitative Results: S3DIS

Method	OA	mAcc	mIoU	door	board
A5 PointNet	–	48.5	41.1	10.7	**26.3**
A5 SEGCloud	–	57.3	48.9	23.1	13.0
A5 SPG	86.4	**66.5**	**58.0**	**61.5**	2.1
PointNet	78.5	66.2	47.6	51.6	29.4
Engelmann	81.1	66.4	49.7	51.2	**30.0**
SPG	**85.5**	**73.0**	**62.1**	**68.4**	8.7

Quantitative Results: S3DIS

Method	OA	mAcc	mIoU	door	board
A5 PointNet	–	48.5	41.1	10.7	**26.3**
A5 SEGCloud	–	57.3	48.9	23.1	13.0
A5 SPG	86.4	**66.5**	**58.0**	**61.5**	2.1
PointNet	78.5	66.2	47.6	51.6	29.4
Engelmann	81.1	66.4	49.7	51.2	**30.0**
SPG	**85.5**	**73.0**	**62.1**	**68.4**	8.7

Śtep	Time
Voxelisation	24
Features	88
Partition	447
SPG computation	436
Inference $\times 10$	60
Total	1055

Superpoint Partition

$$f^\star = \underset{f \in \mathbb{R}^{C \times m}}{\arg\min} \sum_{i \in C} \|f_i - e_i\|^2 + \sum_{(i,j) \in E} w_{i,j} \left[f_i \neq f_j \right],$$

- $e \in \mathbb{R}^{C \times m}$: handcrafted descriptors of the local geometry/radiometry

Superpoint Partition

$$f^\star = \underset{f \in \mathbb{R}^{C \times m}}{\arg\min} \sum_{i \in C} \|f_i - e_i\|^2 + \sum_{(i,j) \in E} w_{i,j} [f_i \neq f_j] \,,$$

- $e \in \mathbb{R}^{C \times m}$: handcrafted descriptors of the local geometry/radiometry
- Superpoints: connected components of a piecewise constant approximation of e structured by an adjacency graph.

Superpoint Partition

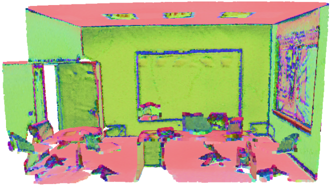
$$f^\star = \underset{f \in \mathbb{R}^{C \times m}}{\arg\min} \sum_{i \in C} \| f_i - e_i \|^2 + \sum_{(i,j) \in E} w_{i,j} \left[f_i \neq f_j \right],$$

- $e \in \mathbb{R}^{C \times m}$: handcrafted descriptors of the local geometry/radiometry
- Superpoints: connected components of a piecewise constant approximation of e structured by an adjacency graph.
- **Problem:** any errors made in the partition will carry in the prediction...

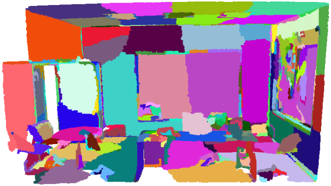
Presentation Layout

The Pipeline

Input Point Cloud



Learned Embedding



Oversegmentation

True Objects

General idea:

1) Train a neural network to produce points embeddings with high contrast at the border of objects...
2) ... Which serve as inputs of a **nondifferentiable** segmentation algorithm.

Adjacency Graph

- $G = (C, E)$ a meaningful adjacency graph

Adjacency Graph

- $G = (C, E)$ a meaningful adjacency graph
- Construction is problem-dependant

Adjacency Graph

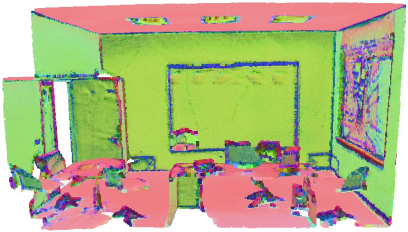
- $G = (C, E)$ a meaningful adjacency graph
- Construction is problem-dependant
- E_{inter} : set of inter-object edges

Adjacency Graph

- $G = (C, E)$ a meaningful adjacency graph
- Construction is problem-dependant
- E_{inter} : set of inter-object edges
- E_{intra} : set of intra-object edges

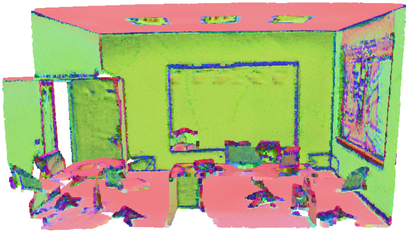
Adjacency Graph

- $G = (C, E)$ a meaningful adjacency graph
- Construction is problem-dependant
- E_{inter} : set of inter-object edges
- E_{intra} : set of intra-object edges
- We want embeddings with high contrast at E_{inter} and similar value at E_{intra}



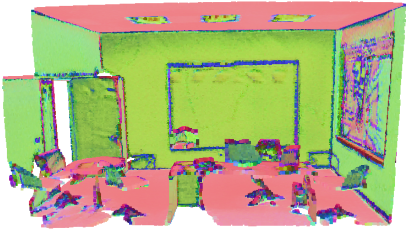
Adjacency Graph

- $G = (C, E)$ a meaningful adjacency graph
- Construction is problem-dependant
- E_{inter} : set of inter-object edges
- E_{intra} : set of intra-object edges
- We want embeddings with high contrast at E_{inter} and similar value at E_{intra}
- If we get E_{inter} right, then we have automatically object purity!



Adjacency Graph

- $G = (C, E)$ a meaningful adjacency graph
- Construction is problem-dependant
- E_{inter} : set of inter-object edges
- E_{intra} : set of intra-object edges
- We want embeddings with high contrast at E_{inter} and similar value at E_{intra}
- If we get E_{inter} right, then we have automatically object purity! almost!



Generalized Minimal Partition Problem

- e_i embeddings of the local geometry/radiometry

Generalized Minimal Partition Problem

- e_i embeddings of the local geometry/radiometry
- **Idea:** Superpoints are the component of a **piecewise-constant approximation** of the embedings

$$f^\star = \underset{f \in \mathbb{R}^{C \times m}}{\arg\min} \sum_{i \in C} \|f_i - e_i\|^2 + \sum_{(i,j) \in E} w_{i,j} \left[f_i \neq f_j\right],$$

Generalized Minimal Partition Problem

- e_i embeddings of the local geometry/radiometry
- **Idea:** Superpoints are the component of a **piecewise-constant approximation** of the embedings

$$f^\star = \operatorname*{arg\,min}_{f \in \mathbb{R}^{C \times m}} \sum_{i \in C} \| f_i - e_i \|^2 + \sum_{(i,j) \in E} w_{i,j} \left[f_i \neq f_j \right],$$

- Superpoints: regions with homogeneous embeddings

Generalized Minimal Partition Problem

- e_i embeddings of the local geometry/radiometry
- **Idea:** Superpoints are the component of a **piecewise-constant approximation** of the embedings

$$f^\star = \underset{f \in \mathbb{R}^{C \times m}}{\arg\min} \sum_{i \in C} \| f_i - e_i \|^2 + \sum_{(i,j) \in E} w_{i,j} \left[f_i \neq f_j \right],$$

- Superpoints: regions with homogeneous embeddings
- Works well with handcrafted embeddings, should work with learned ones!

Generalized Minimal Partition Problem

- e_i embeddings of the local geometry/radiometry
- **Idea:** Superpoints are the component of a **piecewise-constant approximation** of the embedings

$$f^\star = \operatorname*{arg\,min}_{f \in \mathbb{R}^{C \times m}} \sum_{i \in C} \|f_i - e_i\|^2 + \sum_{(i,j) \in E} w_{i,j} \left[f_i \neq f_j\right],$$

- Superpoints: regions with homogeneous embeddings
- Works well with handcrafted embeddings, should work with learned ones!
- **Problem:** a non-convex, nondifferentiable, noncontinuous problem

Generalized Minimal Partition Problem

- e_i embeddings of the local geometry/radiometry
- **Idea:** Superpoints are the component of a **piecewise-constant approximation** of the embedings

$$f^\star = \operatorname*{arg\,min}_{f \in \mathbb{R}^{C \times m}} \sum_{i \in C} \|f_i - e_i\|^2 + \sum_{(i,j) \in E} w_{i,j} \left[f_i \neq f_j\right],$$

- Superpoints: regions with homogeneous embeddings
- Works well with handcrafted embeddings, should work with learned ones!
- **Problem:** a non-convex, nondifferentiable, noncontinuous problem
- Good approximations can be computed with ℓ_0-cut pursuit [Landrieu & Obozinski SIIMS 2018]

The Problem With the GMPP

$$f^\star = \underset{f \in \mathbb{R}^{C\times m}}{\arg\min} \sum_{i\in C} \|f_i - e_i\|^2 + \sum_{(i,j)\in E} w_{i,j}\,[f_i \neq f_j]\,,$$

- Naive approach : loss as the quality of $f^\star$ as a segmentation

The Problem With the GMPP

$$f^\star = \arg\min_{f \in \mathbb{R}^{C \times m}} \sum_{i \in C} \|f_i - e_i\|^2 + \sum_{(i,j) \in E} w_{i,j} \left[f_i \neq f_j\right],$$

- Naive approach : loss as the quality of $f^\star$ as a segmentation
- To backpropagate we need: $\frac{\partial CCC}{\partial f^\star}$ and $\frac{\partial f^\star}{\partial e}$

The Problem With the GMPP

$$f^\star = \underset{f \in \mathbb{R}^{C \times m}}{\arg\min} \sum_{i \in C} \|f_i - e_i\|^2 + \sum_{(i,j) \in E} w_{i,j} \left[f_i \neq f_j\right],$$

- Naive approach : loss as the quality of $f^\star$ as a segmentation
- To backpropagate we need: $\frac{\partial CCC}{\partial f^\star}$ and $\frac{\partial f^\star}{\partial e}$
- **Problem:**Those functions are **not backpropagable**.

Graph-Structured Contrastive Loss

- We propose a *surrogate* loss to learn meaningful embeddings

$$\ell(e) = \frac{1}{|E|}\left(\sum_{(i,j)\in E_{\text{intra}}} \phi\left(e_i - e_j\right) + \sum_{(i,j)\in E_{\text{inter}}} \mu_{i,j}\psi\left(e_i - e_j\right)\right),$$

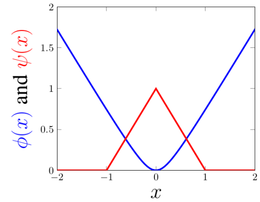
Graph-Structured Contrastive Loss

- We propose a *surrogate* loss to learn meaningful embeddings

$$\ell(e) = \frac{1}{|E|}\left(\sum_{(i,j)\in E_{\text{intra}}} \phi\left(e_i - e_j\right) + \sum_{(i,j)\in E_{\text{inter}}} \mu_{i,j}\psi\left(e_i - e_j\right)\right),$$

- ϕ minimum at 0, ψ maximum at 0

$$\begin{aligned}\phi(x) &= \delta(\sqrt{\|x\|^2/\delta^2 + 1} - 1)\\ \psi(x) &= \max\left(1 - \|x\|, 0\right)\end{aligned}$$



Graph-Structured Contrastive Loss

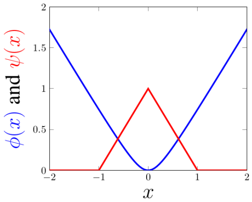
- We propose a *surrogate* loss to learn meaningful embeddings

$$\ell(e) = \frac{1}{|E|}\left(\sum_{(i,j)\in E_{\text{intra}}} \phi\left(e_i - e_j\right) + \sum_{(i,j)\in E_{\text{inter}}} \mu_{i,j}\psi\left(e_i - e_j\right)\right),$$

- ϕ minimum at 0, ψ maximum at 0

$$\begin{aligned}\phi(x) &= \delta(\sqrt{\|x\|^2/\delta^2 + 1} - 1)\\ \psi(x) &= \max\left(1 - \|x\|, 0\right)\end{aligned}$$

- Promotes homogeneity within objects and contrast at their borders



Graph-Structured Contrastive Loss

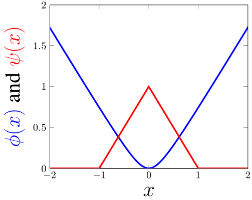
- We propose a *surrogate* loss to learn meaningful embeddings

$$\ell(e) = \frac{1}{|E|}\left(\sum_{(i,j)\in E_{\text{intra}}} \phi\left(e_i - e_j\right) + \sum_{(i,j)\in E_{\text{inter}}} \mu_{i,j}\psi\left(e_i - e_j\right)\right),$$

- ϕ minimum at 0, ψ maximum at 0

$$\begin{aligned}\phi(x) &= \delta(\sqrt{\|x\|^2/\delta^2+1}-1)\\ \psi(x) &= \max\left(1-\|x\|,0\right)\end{aligned}$$

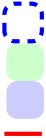
- Promotes homogeneity within objects and contrast at their borders
- $\mu_{i,j}$: weight of inter-edges



Cross-Partition Weighting Strategy, cont'd

$$\mu_{U,V} = \mu \frac{\min(|U|,|V|)}{|(U,V)|} \quad \text{for } (U,V) \in \mathcal{E} \qquad \mu_{i,j} = \mu_{U,V} \text{ for all } (i,j) \in (U,V)$$

- Role of $\mu_{i,j}$ critical: assess impact of missed edge.
- Operate on $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ adjacency graph of cross-partition between superpoints and real objects.



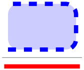
superpoint

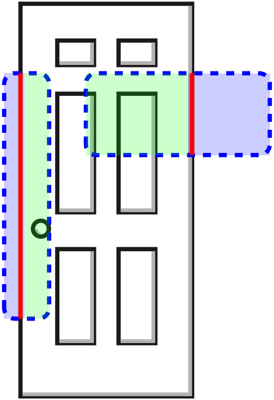
majority object

trespassing

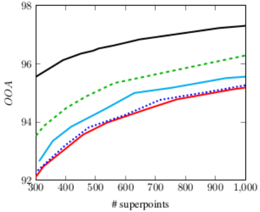
interface

$\mu_{\mathbf{LW},\mathbf{LD}} =$ 

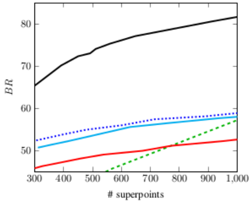
$\mu_{\mathbf{RW},\mathbf{RD}} =$ 



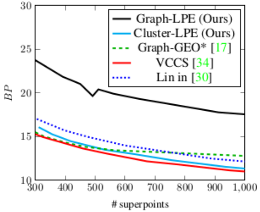
Results



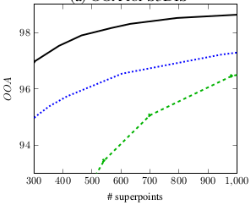
(a) OOA for S3DIS



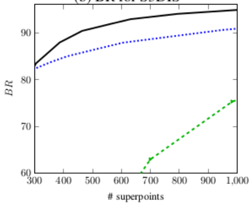
(b) BR for S3DIS



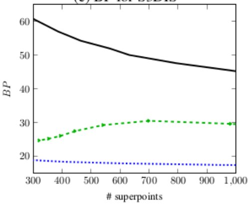
(c) BP for S3DIS



(d) OOA for vKITTI



(e) BR for vKITTI

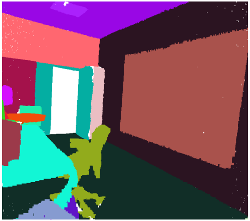


(f) BP for vKITTI

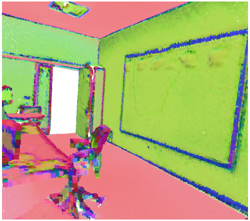
We require 5 times less superpoints for similar performance!

Illustration

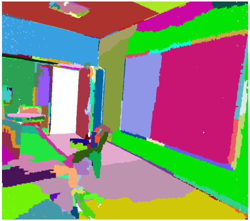
Input cloud



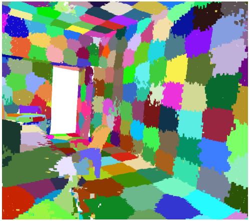
Ground truth objects



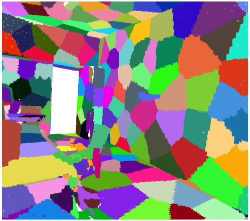
LPE embeddings



Graph-LPE (ours)



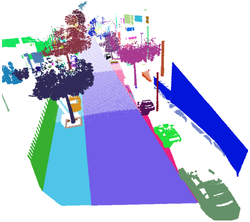
VCCS, Papon *et al.* 2013



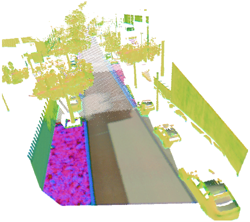
Lin *et al.* 2018

Illustration

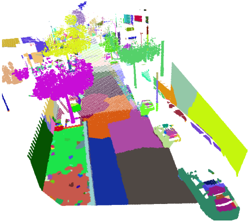
Input cloud



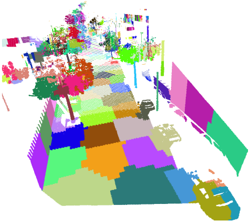
Ground truth objects



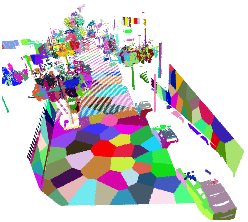
LPE embeddings



Graph-LPE (ours)



VCCS, Papon *et al.* 2013



Lin *et al.* 2018

Results

Method	OA	mAcc	mIoU
6-fold cross validation			
PointNet 2017	78.5	66.2	47.6
Engelmann *et al.* in 2017	81.1	66.4	49.7
PointNet++ 2017	81.0	67.1	54.5
Engelmann *et al.* in 2018	84.0	67.8	58.3
SPG 2018	85.5	73.0	62.1
PointCNN 2018	**88.1**	75.6	65.4
Graph-LPE + SPG (ours)	87.8	**77.5**	**67.6**
Fold 5			
PointNet 2017	-	49.0	41.1
Engelmann *et al.* in 2018	84.2	61.8	52.2
pointCNN 2018	85.9	63.9	57.3
SPG 2018	86.4	66.5	58.0
PCCN 2018	-	67.0	58.3
Graph-LPE + SPG (ours)	**87.8**	**69.1**	**61.5**

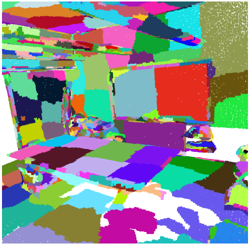
Table: S3DIS

Method	OA	mAcc	mIoU
PointNet 2017	79.7	47.0	34.4
Engelmann 2018	79.7	57.6	35.6
Engelmann 2017	80.6	49.7	36.2
3P-RNN 2018	**87.8**	54.1	41.6
Graph-LPE + SPG (ours)	85.2	**62.4**	**49.7**

Table: vKITTI

Illustration

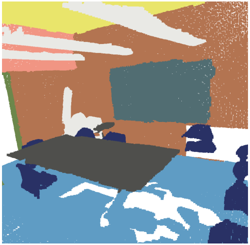
Input Cloud



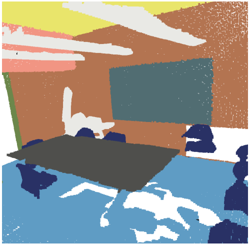
Oversegmentation

S3DIS

- ceiling
- floor
- wall
- column
- beam
- window
- door
- table
- chair
- bookcase
- sofa
- board
- clutter
- unlabelled

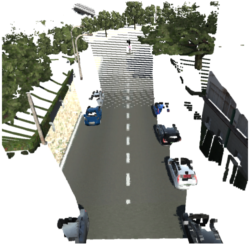


prediction

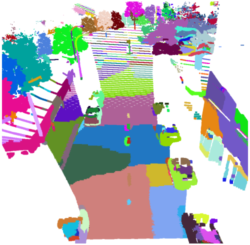


Ground Truth

Illustration



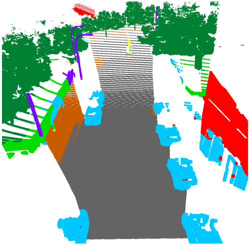
Input Cloud



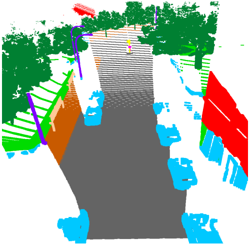
Oversegmentation

vKITTI

- terrain
- tree
- vegetation
- building
- road
- guard rail
- traffic sign
- traffic light
- pole
- misc
- truck
- car
- van
- unlabelled



prediction



Ground Truth

Presentation Layout

Conclusion

- Our paradigm for graph-structured learning and optimization:
- Exploit the spatial regularity of the solution to increase speed and precision.

Conclusion

- Our paradigm for graph-structured learning and optimization:
  - Exploit the spatial regularity of the solution to increase speed and precision.
  - Use neural networks to learn the inputs and parameters of efficient optimization algorithms.

Conclusion

- Our paradigm for graph-structured learning and optimization:
  - Exploit the spatial regularity of the solution to increase speed and precision.
  - Use neural networks to learn the inputs and parameters of efficient optimization algorithms.
  - Use graph-structured optimization to compute the structure of neural network adapted to the data.

Conclusion

- Our paradigm for graph-structured learning and optimization:
  - Exploit the spatial regularity of the solution to increase speed and precision.
  - Use neural networks to learn the inputs and parameters of efficient optimization algorithms.
  - Use graph-structured optimization to compute the structure of neural network adapted to the data.
- All our work is online:
  - `loicland/superpoint-graph` 268 ★ 78
  - `loicland/cut-pursuit` 22 ★ 7
  - `1a7r0ch3/parallel-cut-pursuit` very soon!