Fully Convolutional Architectures for Multi-Part Body Segmentation

Juan Borrego Carazo

University of Barcelona

Fundamentals of Data Science

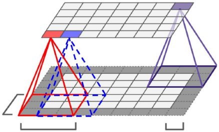
August 27, 2018

Overview

Introduction

Introduction and Background

Mechanism:

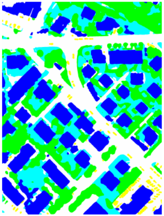


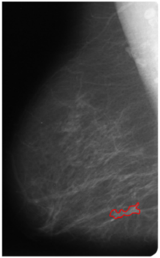
- Appearance of powerful baseline architecture: FCN (Fully Convolutional Network)
- **Task**: semantic segmentation
- Spread of use:
  - Other tasks such as Object Detection: Mask R-CNN
  - Possibility of inclusion in other structures: Encoder-decoders
  - Modification: dilated convolutions
  - Connected to other techniques, such as CRF

Applications

And the reasons behind the spread are?

- Reduction of parameters in networks compared to Fully Connected Networks.
- Excellent feature extractor
- Widespread use in applications and data types:
  - Action recognition
  - Cancer detection
  - Aerial images





Our case & Purpose

Purpose:

- Study the performance and behavior of architectures based fundamentally on convolutions in a specific dataset: SURREAL (Synthetic hUmans foR REal tasks)

Work definition: regarding the nature of our data, the work will be divided in two parts

- General purposed architectures
- Human body specific architectures

Dataset

Dataset

Main characteristics:

- 6.5 million frames grouped into 67582 continuous image sequences of size 320x240 (RGB).
- Synthetic human bodies displayed into a non related background.
- Rich information attached: optical flow, body part segmentation, depth, 3D and 2D joints and surface normals.
- **Body part ground truth segmentation**: 24 body parts each one associated with an integer index (1-24)

Example

Dataset Modifications

Process to obtain final dataset:

- Cut frames and relate them to corresponding GT matrix.
- Crop images with the body on the center.
- Correct GT with parts mislabeled.
- With K-means algorithm create train, validation and test set base on 3D joints information.
- Train: 90k images, Validation: 15k and Test 15k images

Dataset example

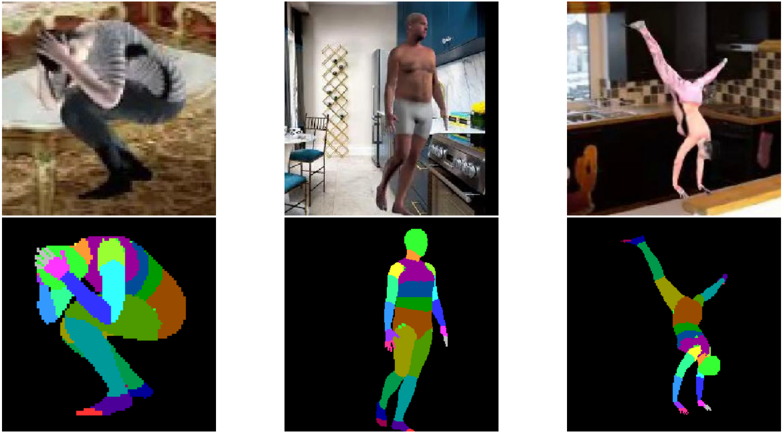


Figure: First row: sample images. Second row: corresponding ground truths

General purposed networks

Experimental Procedure

Take the baseline network and:

- Doubling the convolutional filters
- Data augmentation: mirroring and scaling.
- Class balancing through loss weighting

Class balancing strategy

- **Direct** $L = -\sum_i y_i \log softmax(x_i w_i)$
- **Outter** $L = -\sum_i w_i y_i \log softmax(w_i)$

and weights (C is the number of pixels of each class)

- **Inverse Frequency**: $W_i = 1 - \frac{C_i}{\sum_i C_i}$
- **Exponential weights**:

$$B = \frac{max(C)}{C}$$

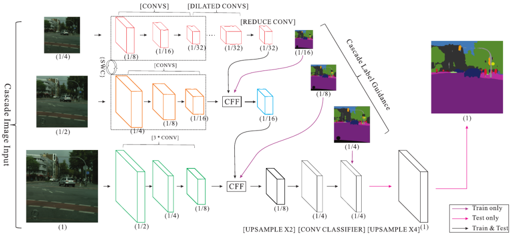
$$W = Be^{-\frac{1}{4}\frac{B-mean(B)}{stdB}}$$

ICNet

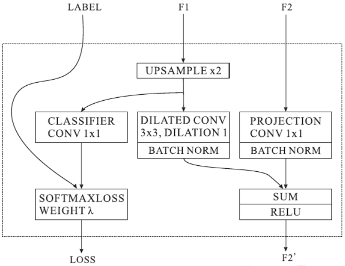
Network Description

General architecture

$$L = \lambda_1 L_1 + \lambda_2 L_2 + \lambda_3 L_3$$



Cascade Feature Fusion



ICNet for Real-Time Semantic

Segmentation on High-Resolution

Images. *Zhao H. et al.*,

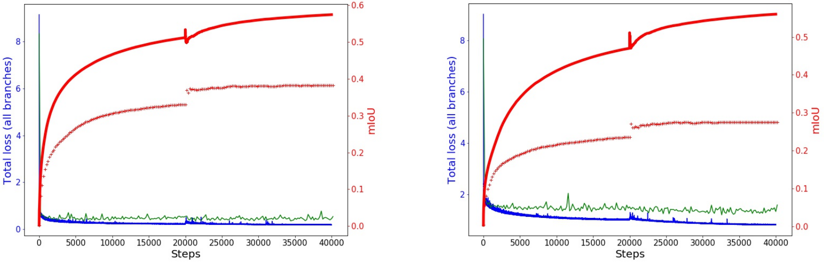
https://arxiv.org/abs/1704.08545

Results and Analysis

Architecture	mIoU (%)	Accuracy (%)	F1 (%)
Normal	38.19	94.64	88.17
Doubled filters	27.51	93.01	84.97
Normal + Data Aug.	32.60	91.15	91.61

Table: Results for the different ablation results in the validation set.

Notice batch normalization inclusion into training.



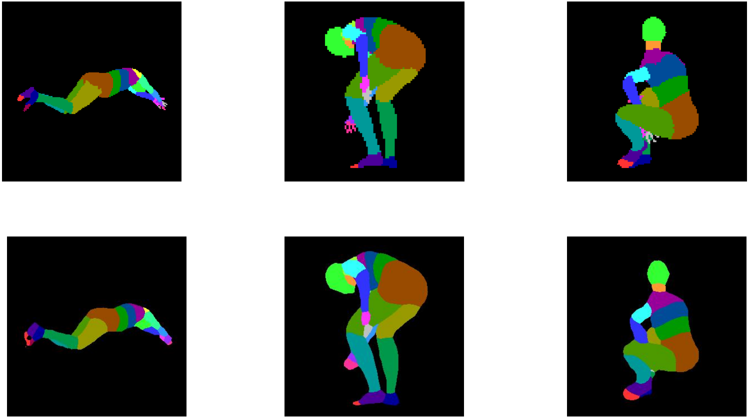
Class balancing results

	mIoU (%)	Accuracy per Class(%)													
Architecture	**All Classes**	**All Classes**	**Background**	**Head**	**Torso**	**U.Legs**	**L.Legs**	**Neck**	**Shoulder**	**U.Arms**	**L.Arms**	**Feets**	**Hands**	**Fingers**	**Toes**
Normal	38.2	48.7	98.9	84.9	74.78	64.3	53.8	64.0	54.2	52.7	39.5	32.3	19.8	9.3	9.5
W1 (Outer)	37.5	52.3	97.7	90.0	74.8	70.9	61.7	60.9	56.0	57.34	50.1	38.9	22.9	10.2	11.3
W1 (Direct)	6.5	7.9	99.9	6.13	15.5	7.7	0.8	0.0	4.7	1.9	0.0	3.6	0.0	0.0	0.0
W2 (Outer)	25.8	54.8	89.2	89.3	61.6	64.1	65.2	72.4	60.3	47.0	46.03	52.35	33.4	31.0	36.9
W2 (Direct)	25.5	34.0	99.3	78.7	70.7	70.0	59.0	1.9	8.9	32.9	15.7	7.1	0.5	0.0	0 .0

Table: Performance results on validation dataset for the original structure and the architecture with loss weighting for each setup. Here W1 indicates the inverse frequency weithing and W2 the exponential weighting. Best values enclosed in [].

Final and Qualitative Results

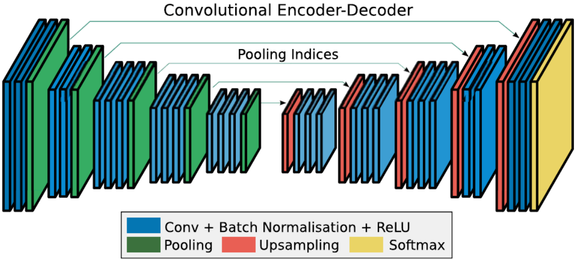
Architecture	mIoU (%)	Accuracy (%)	F1 (%)
Normal	45.14	95.76	89.73



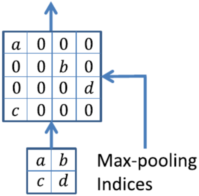
SegNet

Network Description

General architecture



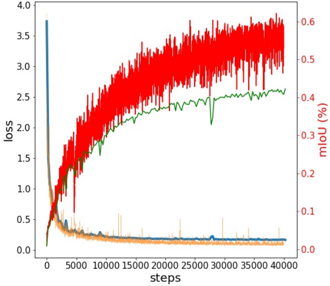
Index Skip connections

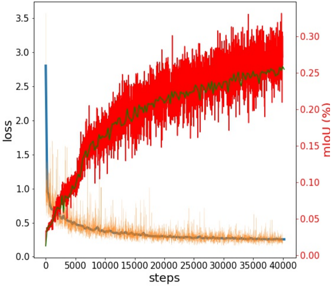


SegNet: A Deep Convolutional Encoder-Decoder Architecture for Robust Semantic Pixel-Wise Labelling. *Badrinarayanan, V.*,https://arxiv.org/abs/1505.07293

Results and Analysis

Architecture	mIoU (%)	Accuracy (%)	F1 (%)
Normal	38.80	94.87	54.34
Doubled filters	39.17	94.79	54.49
Doubled Filters + Data Aug.	23.28	89.24	33.21





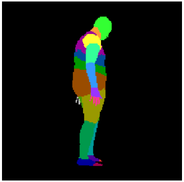
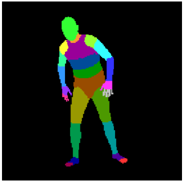
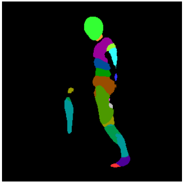
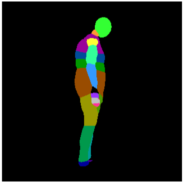
Class balancing results

	mIoU (%)	Accuracy per Class(%)													
Architecture	**All Classes**	**All classes**	**Background**	**Head**	**Torso**	**U.Legs**	**L.Legs**	**Neck**	**Shoulder**	**U.Arms**	**L.Arms**	**Feets**	**Hands**	**Fingers**	**Toes**
Double Filters	39.17	49.9	99.1	84.2	70.9	63.2	58.4	58.1	51.8	52.7	43.9	39.9	28.8	12.4	9.8
DF + W1 (Outer)	38.8	55.6	97.5	90.3	74.2	66.8	61.8	58.3	65.1	62.0	49.6	42.0	36.3	25.3	14.2
DF + W2 (Outer)	21.65	56.3	78.18	79.8	65.6	60.0	57.1	85.8	71.1	52.5	51.8	44.2	41.7	34.1	38.4

Table: Performance results on validation dataset for the doubled filter structure and the same architecture but with loss weighting for each setup. Here W1 indicates the inverse frequency weithing and W2 the exponential weighting (DF, i.e. doubled filters). Between brackets the best perfoming scheme in both mIoU and mean Accuracy per class.

Final and Qualitative Results

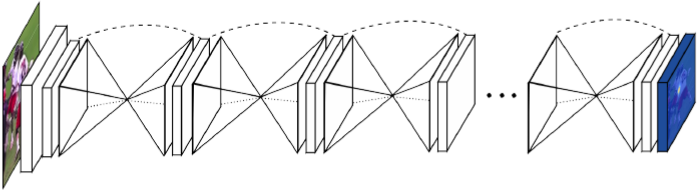
Architecture	mIoU (%)	Accuracy (%)	F1 (%)
Normal	33.59	94.62	44.32


Specific Purpose Network: Stacked Hourglass

Network Description

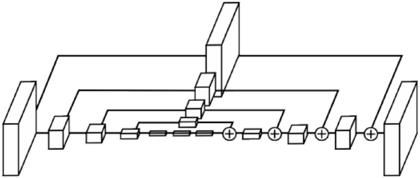
- Originally intended to human pose estimation
- Same bottom-up top-down structure stacked several times
- Allows for refinement of the output produced.



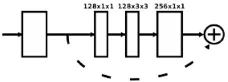
Stacked Hourglass Networks for Human Pose Estimation. *Newell, A.*,https://arxiv.org/abs/1603.06937

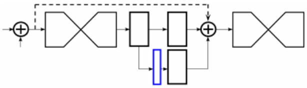
Network Description

Hourglass Module



Residual module & Intermediate Supervision





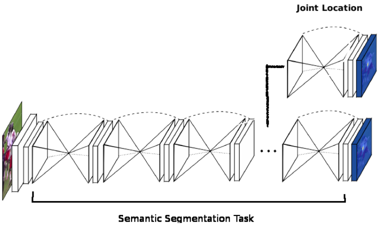
Experimental Procedure

- Two experiments:
  - Different GT resolutions for each intermediate supervision step (i.e. for each hourglass module)
  - A multi-task branch is added to the main pipeline: Joint position determination.

Figure: **1st Experiment**, different ground truth resolutions, one for each module. The idea is to learn a progressive refinement of the real ground truth.

Network Description

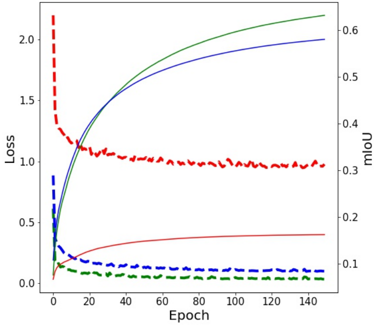
Multi-task branch



Human body joints

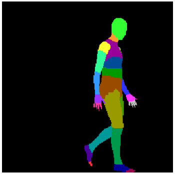
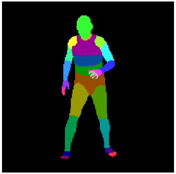
Results and Analysis

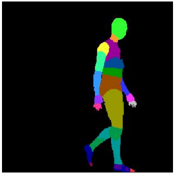
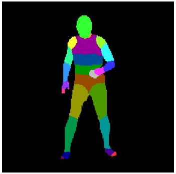
Architecture	mIoU (%)	Accuracy (%)	F1 (%)
Original	63.19	98.75	95.24
O. + GT resolutions	16.22	90.58	61.98
O. + Multitask Head	58.05	97.18	96.05



Final and qualitative results

Architecture	mIoU (%)	Accuracy (%)	F1 (%)
Original	55.32	97.02	93.07

Network Comparison

Test and qualitative results

Architecture	mIoU (%)	Accuracy (%)	F1 (%)
ICNet	45.14	95.76	89.73
SegNet	33.59	94.62	44.32
Stacked Hourglass	55.32	97.02	93.07

- Differences between networks:
  - **ICNet**, 3 branches different resolutions. Only upper branch used in testing. 6,743,733 trainable variables.
  - **SegNet**, encoder-decoder with skip connections. 5,904,921 trainable variables.
  - **Stacked Hourglass**: concatenated downsampling-upsampling with residual modules. 14,804,962 trainable variables.
- Raises the following question: which is the reasons behind the difference in performance:
  - Size of network?
  - Suitability to data type?

Qualitative Results

GT

ICNet

SegNet

Hourglass

Conclusions

Conclusions and future work

Conclusions

- Three CNN analyzed: two general purposed and one data suited network. Acquired level of Tensorflow to modify adapt state of the art code.
- Almost all ablation experiments and modifications did not overperformed original networks.
- Main drawback of study: is it size or network specialization that drives performance?

Future work

- Include more networks
- Adapt network parameters or size to make them comparable.

The End