

UNIUNEA EUROPEANĂ





Instrumente Structurale 2014-2020

Efficient Deep Learning in CloudUT



Raluca Brehar, Ion Giosan, Cristian Vancea Computer Science Department Technical University of Cluj-Napoca Raluca.Brehar@cs.utcluj.ro Ion.Giosan@cs.utcluj.ro Cristian.Vancea@cs.utcluj.ro





Computer Vision Tasks



source: http://cs231n.stanford.edu/slides/2020/lecture 12.pdf

Solutions: Approaches with a high level of complexity based on neural networks.



Objectives



- Usage of the CloudUT infrastructure for applications that need:
 - High GPU processing for deep learning problems
 - Large storage space (deep learning applications need large datasets of annotated images in order to train accurate models).
- > Preparation of the application on the local workstation
 - If the necessary resources are not enough, the CloudUT machines can be used.
- ➤ The application is ported on CloudUT (solves problems from a certain level of complexity → allows scaling for complex problem solving)
- We have experimented with PyTorch and MATLAB deep learning applications.





Develop and run the application on the local machine

 Model design, establish training parameters: epochs, batch size, learning rate, other hyper-parameters

Application porting in CloudUT infrastructure

- 1. Request resources in CloudUT
- 2. Based on the request analysis the system engineer provides a virtual machine which is accessible by VPN
- 3. Copy the data for the application into CloudUT infrastructure (FTP, RDP)
- 4. Run the application in cloud (allows scalability for complex problem solving)



MATLAB

- Provides support for signal processing, optimization of functions, control system design, image and audio processing, machine learning and deep learning
- Needs high computational resources for parallel and distributed computing: memory, CPU, GPU
- Popular in the scientific community
- License based



PyTorch application development

PyTorch

- Open-source availability
- Flexibility
- Distributed model parallel training
- Mobile support
- Robust ecosystem an active community of researchers and developers have built a rich ecosystem of tools and libraries for extending PyTorch and supporting development in areas from computer vision to reinforcement learning
- Native ONNX support
- Cloud support





Deep learning in MATLAB and PyTorch **Cloud**

- Support for convolutional neural network design, training and for prediction based on the trained models
- Training/prediction processes are scalable with respect to available computing resources
- The speed increase is proportional to the work capabilities of the machine used for training



Purpose and advantages of working with deep learning models in CloudUT



- For research teams
- Enables the efficient execution of applications
- Usage of CloudUT infrastructure configured to each application needs
- Reduces the execution time
- Possibility to test applications not running on personal computers limited by resources or insufficient storage space.
- Quick access to TUCN network & privacy





1) Handwritten digit recognition 2) Semantic segmentation of color images







1. Digit recognition



• CNN architecture





ICCP 2021

1. Digit recognition



Dataset: MNIST (28x28x1)
 <u>https://en.wikipedia.org/wiki/MNIST_database</u>

- Hyper-parameters
 - Number of convolutional filters
 - Number of epochs
 - Batch size





Local machine configuration

- Ubuntu 18.04
- GPU: NVIDIA GeForce RTX 2080 Ti/PCle/SSE2 with 11GB memory
- Processor: Intel i7-3770K CPU 3.5GHz (4 processing cores and 8 virtual cores) + 16GB RAM

Virtual machine configuration in CloudUT

- Ubuntu 20.04
- GPU: NVIDIA V100Q with 32GB memory
- Processor Intel Xeon Gold 6230 2.1GHz (8 processing cores) + 128GB RAM



1. Digit recognition in PyTorch



Local Machine											
Epochs	Batch Size	Training Time	Accuracy								
50	128	0:04:56	87 %								
50	256	0:04:36	86%								
50	512	0:04:26	81%								
CloudUT											
50	128	0:04:29	88%								
50	256	0:04:19	85%								
50	512	0:04:14	78%								

NO noticeable increase in accuracy or execution time ! Proves the Cloud infrastructure is suitable for high computation applications.

https://medium.com/mini-distill/effect-of-batch-size-on-training-dynamics-21c14f7a716e





Test configurations

- 1. Local machine: Windows 10
 - **GPU** NVIDIA GeForce RTX 2080 Ti/PCIe/SSE2 12GB memory
 - CPU Intel i7-3770K@3.5GHz (8 cores)
 - 16GB **RAM**
- 2. Virtual machine 1, in CloudUT: Windows 10
 - **GPU** NVIDIA V100 with 16GB memory
 - **CPU** Intel Xeon Gold 6230@2.1GHz (8 processing cores)
 - 32GB **RAM**
- 3. Virtual machine 2, in CloudUT: Windows 10
 - GPU NVIDIA V100 with 32GB memory
 - **CPU** Intel Xeon Gold 6230@2.1GHz (8 processing cores)
 - 128GB **RAM**



1. Digit recognition in MATLAB



Training session in CloudUT with available resources and their usage on virtual machine 1

🔺 MATLAB R2020b - acaden	mic use									- 🗆 ×	🔒 Device M	lanager — 🗆 🗙
HOME PLOTS	A	PPS I	EDITOR	PUBLISH VIEW			82583) e e (Search Docum	ientation 🔎 🐥 Sign	n File Action	n View Help
New Open Save	nd Files	Go To ♥ Go To ♥ Go Find ♥ NAVIGATE	Insert 🛃 fx comment % % Indent 🛐 🛃 EDIT	Breakpoints BREAKPOINTS	Run Run and Advance	Run Section Advance Run and Time					← → □ ✓ ∴ Matlak > ↓ Au > ≫ Bat > □ Co	b-GPU dio inputs and outputs tteries mputer
Current Folder	C: • Work •	MATLAS B	ASICCININ .	Paris CNN m				0.0	Worksonge		> Dis	sk drives splav adapters
Current Folder () Name A saved-models tuinBasicCNN.m	Editor - ↓ trainBa 52 53 54 55 55 56 57 58 - 59 60 61 62 63 Command I I	Editor - C:Work/MATLABBasicCNN.m tromBasicCNN.m * + fullyConnectedLayer (numClasses) softmaxLayer classificationLayer]; options = trainingOptions('adam', 'ValidationData', imdyValidation, 'ValidationFrequency',10, 'InitialLearnRate', 0.01, 'LearnRateSchedule', 'piecewise', 'LearnRateSchedule', 'Diecewise', 'LearnRateSchedule', 0.8, mmmandWindow 46 1130 00:01:31 100.00% 98.94% 0.00 46 1140 00:01:32 100.00% 99.93% 0.00						0031 0.14	Name A accuracy add path epochs infle.ext imdsTrain infle.ext indsTrain infle.ext indsTrain infle.ext indsTrain infle.ext indsTrain infle.ext indsTrain infle.ext indsTrain infle.ext indsTrain infle.ext indsTrain infle.ext infle.ext indsTrain infle.ext infl	Value 0.9992 '\Data\digits\' 50 '.png' [28,28,1] ht imageDatastore ht imageDatastore ht imageDatastore ht imageDatastore ht struct './saved-models/ac_1_info.ma fish Layer ht SeriesNetwork './saved-models/ac_1_net.mal 10 '\Data\digits\test\' '\Data\digits\test\' '\Data\digits\test\' '\Data\digits\test\'	2	Spay adopters Microsoft Remote Display Adapter (MVDIA GRID V100D-16Q VMware SVGA 30 U/CO-R-ROM drives uman Interface Devices E ATA/ATAPI controllers E ATA/ATAPI controllers Spoards ice and other pointing devices onitors tevork adapters int queues occessors flyware devices orage controllers stem devices inversal Senial Bus controllers
		1 47 1160 00:01:33 100.008 99.008 1 47 1170 00:01:34 100.008 99.998 48 1180 00:01:35 100.008 98.998 48 1190 00:01:35 100.008 98.938 48 1190 00:01:36 100.008 98.938 48 1200 00:01:36 100.008 99.938 49 1220 00:01:36 100.008 99.928 50 1220 00:01:38 100.008 99.928 50 1230 00:01:39 100.008 99.928 50 1240 00:01:39 100.008 98.978 50 1240 00:01:39 99.958 98.974	99.00% 90.99% 90.93% 98.93% 99.02% 98.92% 98.92% 98.92% 98.92%	0.0025 0.0027 0.0022 0.0029 0.0028 0.0027 0.0028 0.0028 0.0033 0.0034	0.4 0.4 0.4 0.4 0.4 0.4 0.4	predictedLabels predictedLabels predictedScores	100000 File Options View 100000 File Options View Processes Performa CPU 6% 2;	nce App history Sta 0 GHz OTV	App history Startup Users Details Services CPU Intel(R) Xeon(R) Gold 6230 CPU @ 2.10Gi % Utilization over 60 seconds 10			
trainBasicCNN.m (Sc A	1 1	acy = .9892	1250	00:01:40 1	39.958 [95.946	0.0031 1	0.1 	<	7,0/32 Disk 0% Ether \$168 St 168	0 GB (22%) 0 GC:) et0 2 R: 40.0 Kbps 0 4 GRID V100D-16Q	Utilization Speed Base speed: 2.10 GHz 6% 2.10 GHz Sockets: 2 Virtual processor: 8 Processor Threads Handles Virtual processor: 8 149 1920 100199 L1 cache N/A Up time 5:04:37:46
										Fewer details	Open Resource Mo	i onitor

1. Digit recognition in MATLAB



Training session in CloudUT with available resources and their usage on virtual machine 2

ONE PUT DOTE DOTE <thdote< th=""> DOTE DOTE</thdote<>	📣 MATLAB R2020b - acad	emic use										-		🙀 Task Ma	nager			-	
Image: Window	HOME PLOTS	ş ()	APPS	EDITOR	PUBLISH	VIEW			11 1 9 6	000	Search Doo	umentation	🔎 🌲 Sign In	Eile Option	s ⊻iew				
	👍 🛌 🖾 🖓 A	ind Files	44	Insert 🔜 f	x 🖪 🕶 👔	1			1					Processes P	erformance App history Start	up Users Details Service	5		
• • • • • • • • • • • • • • • • • • •	New Open Save D C	ompare 👻	Go To •	Comment % 3	Break	kpoints Ru	n Run and Advance	Advance	Run and Time				Ā	·	CPU 4% 2.09 GHz	GPU		NVIDIA GRID V	/100D-32Q
Dument for the image of the concentration days of the concentrati	🗘 🔶 🛅 🌄 🖾 🖉 🛑 🗸	C: • Work •	MATLAB + I	BasicCNN 🕨									م •		Memory	~ Cuda	0% *0	эру	0%
Image Image <th< td=""><td>Current Folder</td><td>Edito</td><td>r - C:\Work\MA</td><td>TLAB\BasicCNN\tr</td><td>ainBasicCNN.m</td><td></td><td></td><td></td><td></td><td></td><td>⊙ ×</td><td>Workspace</td><td>۲</td><td></td><td>15/128 GB (12%)</td><td></td><td></td><td></td><td></td></th<>	Current Folder	Edito	r - C:\Work\MA	TLAB\BasicCNN\tr	ainBasicCNN.m						⊙ ×	Workspace	۲		15/128 GB (12%)				
Bundle accoder Bundl	Name *	sem	Seg.m 🛛 tri	ainBasicCNN.m	(+						_	Name *	Value						
Command Window i	aswed-models MarinBasicCNN.m	55 56 57 59 60 61 62 63 64 65 66 65 66 67 68 69 70 71 72 - <	class options %decoment % [net, inf	<pre>= trainingOp 'Valid 'Valid 'Valid 'Initia 'Learn 'Learn 'Learn 'AsxEp 'Minin 'Shuff 'Shuff 'Executation cati daca se enfo] = trainNe </pre>	rer]; htions('adam lationData', lationFreque llearnRate' RateSchedul RateDropPer wochs', epoc latchSize', 'le', 'every ttionEnviron doreste uti Network(imdST	i, imdsValida ncy',10, . , 0.01, e', 'piece tor', 0.8, iod', 10, hs, 32769, y-epoch', ment', 'gp llizarea au imdsTrain, train, layer	<pre>wise',</pre>	ons);			× *	accuracy data_path eccuracy data_path epoth epoth imageSize imageSize imdsTest imdsTest imdsTest imdsTest imdsValidation info info	0.9628 :\.\Data\digits\' 50 20,000 Izng' (28,26,1) Ix1 ImageDataston Ix1 ImageDataston Ix1 ImageDataston Ix1 ImageDataston Ix1 Isruet '/saved-models/a- ISr1 Layer Ix1 SeriesNetwork '/saved-models/a- Ix1 SeriesNetwork '/saved-models/a- Ix1 TrainingOption '.\.Data\digits\tri .\.Data\digits\tri .\.Data\digits\tri .\.Data\digits\tri .\.Data\digits\tri Data\digit	humber	Disk 0 (C:) 2% Ethernet Ethernet0 2 5: 168 R: 24.0 Kbps GPU 0 NVIDIA GRID V100D-32Q 0%	Video Encode Dedicated GPU memory u	0% ~Vi	deo Decode	0% 0% 300 GB
I Epoch Teration Time Elapsed Mini-batch Validation Mini-batch Valida / Loss		Comman	d Window								۲	ing testabels	Tootox r Eulegonice						
 1 1		1	Spoch 1 	Iteration 	Time Elaps (hh:mm:ss	sed Mir s) Ac	ni-batch ccuracy	Validation Accuracy	Mini-b Los	atch s	Valida ^ Los					Shared GPU memory usa	ge		64.0 G8
fx x x x Details A A A A III- UTF-8 script Ln 63 Col 31		 	1 10 20 30 40 50 11 10 20 30 40 50 11 10 20 30 40 50 5	1 10 20 30 40 50	00:00 00:00 00:00 00:00 00:00	0:02 0:11 0:21 0:30 0:40 0:50	11.50% 75.88% 90.39% 93.95% 95.33% 96.15%	27.79 81.37 91.70 94.62 95.39 96.13	% 2 % 0 % 0 % 0 % 0 % 0 % 0	.8301 .6876 .2987 .2000 .1520 .1280	4 0 0 0 0					Utilization 0% GPU Memory 14.3/94.0 GB	Dedicated GPU memor 14.2/30.0 GB Shared GPU memory 0.1/64.0 GB	y Driver version: Driver date: DirectX version: Physical location: Hardware reserved	2 1 P memory: 1
Details C C C UTF-8 script Ln 63 Col 31		fx; >>									~								
	Details ^	<						UTF-8	script		>	< Ln 63	Col 31	S Fewer s	details 🔊 Open Resource Mo	onitor			





Comparative results

	Parameters		Executi worl	on on local station	Execution machine ₁	on virtual (CloudUT)	Execution on virtual machine ₂ (CloudUT)		
Epochs	lmage size	Batch size	Accuracy	Training time (hh:mm:ss)	Accuracy	Training time (hh:mm:ss)	Accuracy	Training time (hh:mm:ss)	
50	28x28x1	1024	0.9899	0:05:42	0.9887	0:02:39	0.9898	0:02:44	
50	28x28x1	2048	0.989	0:04:22	0.9892	0:01:40	0.989	0:01:39	
50	28x28x1	4096	0.9898	0:03:46	0.99	0:01:13	0.9898	0:01:12	
50	28x28x1	8192	0.9876	0:03:29	0.9875	0:01:07	0.9875	0:01:00	
50	28x28x1	16384	0.9832	0:03:37	0.9829	0:00:58	0.9829	0:01:00	
50	28x28x1	32768	0.9628	0:03:32	0.9628	0:01:06	0.9628	0:00:50	





Comparative analysis of execution time

- On virtual machine 1 from CloudUT (compared with the local machine) there is a mean decrease of the execution time of 65% (min. 53%, max. 73%)
- On virtual machine 2 from CloudUT, with a large batch size (32 768 images), there is a time execution decrease of 24% with respect to the virtual machine 1
 - Due to the high processing power of the GPU for a large batch size
 - The computations were made faster due memory increase (32GB for virtual machine 2 vs. 16GB for virtual machine 1)



2. Semantic segmentation in PyTorch **Cloud**

• ERFNet[1] network



(1) Encoder: Layers 1-16

- Residual and downsampling blocks + interleaved dilated convolutions
- (2) Decoder: Layers 17-23
- Up-samples the feature maps to match the resolution of the deconvolution layers.



1024x512x3

(RGB IMAGE)

2. Semantic segmentation in PyTorch **Cloud**

- ERFNet[1] trained on Cityscapes[2]
 - Cityscapes dataset:

- Urban images containing 30 semantic classes with pixel level annotations.

- 5,000 annotated images
- Hyper-parameters:
 - Epochs: 50, 100
 - Batch size: 2, 3, 4, 8
 - Image size: 512x1024



2. Semantic segmentation in PyTorch **Cloud**

Local machine results (images of size 512x1024)											
Epochs	Batch Size	Encoder training	Decoder training	IoU on VAL	Average time per image						
50	2	2h:27m:55s	2h:55m:55s	65.44 %	59ms						
50	3	2h:15m:25s	2h:40m:42s	66.29 %	56ms						
50	4	RuntimeError:	CUDA out of memory.	*							
Virtual m	achine from	CloudUT results (ir	nages of size 512x10	24)							
50	2	2h:58m:43s	3h:08m:54s	65.89%	68ms						
50	3	2h:48m:43s	2h:48m:43s	66.58%	63ms						
50	4	2h:04m:40s	2h:30m:43s	66.24%	60ms						
50	6	1h:29m:57s	2h:19m:33s	68.43%	45ms						
50	8	1h:15m:57s	2h:09m:53s	66.35%	28ms						

A speed increase factor of 1.4 while training in Cloud, for a large batch size, increase in accuracy !

For a small batch size the accuracy is comparable on the local machine and on the virtual machine.

*Tried to allocate 32.00 MiB (GPU 0; 10.76 GiB total capacity; 9.19 GiB already allocated; 41.81 MiB free; 68.85 MiB cached)





Deep learning architecture

DeepLabv3+ [1]: Encoder (Resnet-18) + Decoder

- Layers: Convolution, Batch Normalization, ReLU, Image Pooling
- Number of layers: 100
- Resnet-18 [2] (1000 classes): pre-trained on 10⁶ images from Imagenet (source: http://www.image-net.org)





CamVid dataset (701 images 960x720x3) – 32 classes

Source: http://web4.cs.ucl.ac.uk/staff/g.brostow/MotionSegRecData



Original image

Annotated image



ICCP 2021

Training parameters

- Stochastic Gradient Descent with Momentum
- Learning rate: 0.003 (initial), decay with a factor of 0.3 every 10 epochs.
- Number of epochs: variable
- Batch size: variable

Evaluation metrics

- Accuracy
- Intersection over Union (IoU)
- F1 score on contours (BFScore)





Test configurations

- 1. Local workstation: Windows 10
 - GPU NVIDIA GeForce RTX 2060 SUPER 8GB memory
 - CPU Intel i7-3770K@3.5GHz (8 threads)
 - 16GB **RAM**
- 2. Local server: Windows 10
 - GPU NVIDIA GeForce GTX 1080Ti with 12GB memory
 - **CPU** Intel i9-7900X@4GHz (20 processing cores)
 - 128GB **RAM**
- 3. Virtual machine 1, in CloudUT: Windows 10
 - **GPU** NVIDIA V100 with 16GB memory
 - CPU Intel Xeon Gold 6230@2.1GHz (8 processing cores)
 - 32GB **RAM**





Experimental results (virtual machine 1, CloudUT)



Batch Size = 8, Training: 1550 iterations (30 epochs)



ICCP 2021

Experimental results (virtual machine 1, CloudUT)

					Testing partition			
Batch Size Training: 1	e = 8 1550 iterat	ions (30 e	epochs)	Class	loU	Accuracy	BFScore	
				Sky	0.90804	0.93887	0.90636	
Global accuracy	Average accuracy	Average IoU	Weighted IoU	Average BFScore	Building	0.80341	0.82986	0.66167
0.89658	0.86185	0.66822	0.83346	0.70117	Pillow	0.25494	0.74593	0.59589
					Road	0.93057	0.94582	0.81708
Batch Siz	e > 8 => <mark>E</mark> I	rror: insuf	fficient RAM	l	Pavement	0.7472	0.8924	0.76095
					Tree	0.77623	0.88676	0.73405
					Sign symbol	0.43684	0.75522	0.55289
					Fence	0.59367	0.81132	0.59107
					Car	0.79434	0.92213	0.75142
Remark: I	ower effic	iency for	small size o	biects	Pedestrian	0.4633	0.86498	0.6267
					Bicyclist	0.64193	0.88703	0.5652





Comparative results

	Para	ameters		Execution machine	n on virtual (CloudUT)	Executio se	on on local rver	Execution on local workstation		
Epochs	Iterations	lmage size	Batch size	Accuracy	Time (hh:mm:ss)	Accuracy	Time (hh:mm:ss)	Accuracy	Time (hh:mm:ss)	
1	1	960x720x3	8	0.0721	00:00:45	0.0966	00:00:26	0.0721	00:00:49	
2	104	960x720x3	8	0.8375	00:07:44	0.8366	00:06:59	0.8364	00:16:07	
4	208	960x720x3	8	0.8475	00:14:43	0.8463	00:13:34	0.8468	00:31:01	
8	400	960x720x3	8	0.9042	00:27:34	0.8871	00:25:21	-	-	
16	800	960x720x3	8	0.9178	00:55:12	0.9177	00:51:30	1	1	
25	1300	960x720x3	8	0.9244	01:27:38	0.9246	01:23:15	-	-	
30	1550	960x720x3	8	0.9079	01:43:53	0.9083	01:39:09	-	-	

Remark: $P_{virtual m1} \approx P_{server} = 2.2 \times P_{workst}$ Insufficient resources

CloudUT,



Test configurations

- 1. Local server: Windows 10
 - GPU NVIDIA GeForce GTX 1080Ti with 12GB memory
 - CPU Intel i9-7900X@4GHz (20 processing cores)
 - 128GB **RAM**
- 2. Virtual machine 2, in CloudUT: Windows 10
 - **GPU** NVIDIA V100 with 32GB memory
 - **CPU** Intel Xeon Gold 6230@2.1GHz (8 processing cores)
 - 128GB RAM





Time and resource during training (max. 30 epocs)

Batch	Necessary	Clo - virtua	oudUT I machine 2 -	Loc	al server	CloudUT (max values) - virtual machine 2 -			
size	size iterations	Time (hh:mm)	Performance (%)	Time (hh:mm)	Time Performance (hh:mm) (%)		GPU Mem (GB)	RAM (GB)	
2	1300	00:29	355	00:26	396	30	8	30	
4	1700	00:51	202	01:00	172	30	8	30	
6	1750	01:05	158	01:24	123	50	11	33	
8	1550	01:14	139	01:39	104	60	12	35	
10	1000	01:05	158	01:31	113	62	14	35	
12	1050	01:22	126	04:00	43	68	16	37	
14	900	01:24	123	03:13	53	70	19	37	
16	780	01:24	123	03:10	54	73	20	38	

Performance = Time / Time_{virtual_m1(batch size=8)}



Comparative analysis

MATLAB uses efficient parallelization algorithms => performance scalability with respect to available resources

- Small Batch Size -> reduced GPU consumption (memory/CUDA)
 => speed performance increase up to 150% for virtual machine 2.
- Large Batch Size -> high GPU consumption (memory/CUDA)
 => speed performance increase up to 230-300% for virtual machine 2
- Parallel computing made on GPU allows a stabilization of the working time for large batch size with respect to small batch size, when there are enough computing resources.
- For complex models we recommend virtual machines with at least 32G RAM on GPU.





CloudUT advantages for deep learning

- Decreased execution time depending on the application complexity.
- Allows larger batch size that can impact an increase in accuracy.

Steps for developing an application on CloudUT

- 1. Develop the application on the local machine.
- 2. Establish what are the necessary hardware resources for a more efficient training.
- Ticketing request for the CloudUT administrator => provides a virtual machine.
- 4. Install on the virtual machine all the needed libraries for your application. Port the necessary data.
- 5. Run the application on the provided virtual machine. Save the models!



Bibliography



- [1] "ERFNet: Efficient Residual Factorized ConvNet for Real-time Semantic Segmentation", E. Romera, J. M. Alvarez, L. M. Bergasa and R. Arroyo, Transactions on Intelligent Transportation Systems (T-ITS), [Accepted paper, to be published in Dec 2017].
- [2] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.





UNIUNEA EUROPEANĂ





Thank you !



Raluca Brehar, Ion Giosan, Cristian Vancea Computer Science Department Technical University of Cluj-Napoca

Raluca.Brehar@cs.utcluj.ro Ion.Giosan@cs.utcluj.ro Cristian.Vancea@cs.utcluj.ro