Article

Intelligent Computer-Aided Model for Efficient Diagnosis of Digital Breast Tomosynthesis 3D Imaging Using Deep Learning

Alaa M. Adel El-Shazli [1,] *, Sherin M. Youssef [2] and Abdel Hamid Soliman [3]

[1] Computer Engineering Department, Arab Academy for Science, Technology and Maritime Transport (AASTMT), Giza 12577, Egypt; alaa.alshazli@aast.edu (A.E.-S.)
[2] Computer Engineering Department, Arab Academy for Science, Technology and Maritime Transport (AASTMT), Alexandria 1029, Egypt; sherin@aast.edu (S.M.Y.)
[3] School of Digital, Technologies and Arts, Staffordshire University, Stoke-on-Trent ST4 2DE, UK; A.Soliman@staffs.ac.uk (A.A.)

* Correspondence: alaa.alshazli@aast.edu; Tel.: +201287278224

Table S1. ResNet-18's architectural details [16]

Layer Name	Output Size	ResNet-18
conv1	$112 \times 112 \times 64$	7×7, 64, stride 2
conv2_x	$56 \times 56 \times 64$	3×3 max pool, stride 2
		$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$
conv3_x	$28 \times 28 \times 128$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$
conv4_x	$14 \times 14 \times 256$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$
conv5_x	$7 \times 7 \times 512$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$
average pool	$1 \times 1 \times 512$	7×7 average pool
fully connected	1000	512×1000 fully connections
softmax	1000	

Table S2. AlexNet's architectural details [20]

Layer		Feature Map	Size	Kernel Size	Stride	Activation
Input	Image	1	227x227x3	-	-	-
1	Convolution	96	55 x 55 x 96	11x11	4	relu
	Max Pooling	96	27 x 27 x 96	3x3	2	relu
2	Convolution	256	27 x 27 x 256	5x5	1	relu
	Max Pooling	256	13 x 13 x 256	3x3	2	relu
3	Convolution	384	13 x 13 x 384	3x3	1	relu
4	Convolution	384	13 x 13 x 384	3x3	1	relu
5	Convolution	256	13 x 13 x 256	3x3	1	relu
	Max Pooling	256	6 x 6 x 256	3x3	2	relu
6	FC	-	9216	-	-	relu
7	FC	-	4096	-	-	relu
8	FC	-	4096	-	-	relu
Output	FC	-	1000	-	-	Softmax

Table S3. GoogleNet layers [22]

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

Table S4. MobileNetV2 layers [26]

Input	Operator
$224^2 \times 3$	conv2d
$112^2 \times 32$	bottleneck
$112^2 \times 16$	bottleneck
$56^2 \times 24$	bottleneck
$28^2 \times 32$	bottleneck
$14^2 \times 64$	bottleneck
$14^2 \times 96$	bottleneck
$7^2 \times 160$	bottleneck
$7^2 \times 320$	conv2d 1x1
$7^2 \times 1280$	avgpool 7x7
$1 \times 1 \times 1280$	conv2d 1x1

Table S5. Comparison between the six DL models on Dataset 1 after pre-processing in terms of Accuracy with a 95% Confidence Interval (CI) and Run-time in seconds

Deep Learning Model	Accuracy (± 95% CI)	Run-time (sec)
ResNet-18	48.65% (±6.93%)	649
AlexNet	56.86% (±6.86%)	652
GoogleNet	47.60% (±6.92%)	1,141
VGG 16	50.67% (±6.93%)	4,759
MobileNetV2	51.53% (±6.93%)	780
DenseNet-201	46.11% (±6.91%)	4,479

Table S6. Comparison between the six DL models on Dataset 2 in terms of Accuracy with a 95% Confidence Interval (CI) and Run-time in seconds

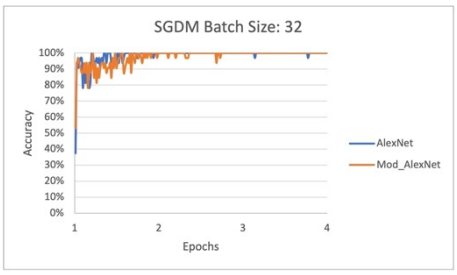
Deep Learning Model	Accuracy (± 95% CI)	Run-time (sec)
ResNet-18	67.43% (±5.30%)	1,535
AlexNet	77.80% (±4.70%)	1,040
GoogleNet	63.08% (±5.46%)	1,986
VGG-16	68.89% (±5.24%)	9,174
MobileNetV2	68.42% (±5.26%)	1,663
DenseNet-201	65.40% (±5.38%)	9,259

Table S7. Comparison between the six DL models on Dataset 1 and Dataset 2 in terms of Accuracy with a 95% Confidence Interval (CI) and Run-time in seconds

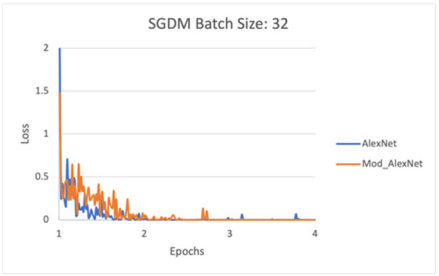
Deep Learning Model	Dataset 1		Dataset 2	
	Accuracy (± 95% CI)	Run-time (sec)	Accuracy (± 95% CI)	Run-time (sec)
ResNet-18	48.65% (±6.93%)	649	67.43% (±5.30%)	1,535
AlexNet	56.86% (±6.86%)	652	77.80% (±4.70%)	1,040
GoogleNet	47.60% (±6.92%)	1,141	63.08% (±5.46%)	1,986
VGG-16	50.67% (±6.93%)	4,759	68.89% (±5.24%)	9,174
MobileNetV2	51.53% (±6.93%)	780	68.42% (±5.26%)	1,663
DenseNet-201	46.11% (±6.91%)	4,479	65.40% (±5.38%)	9,259

Table S8. Comparison between the six DL models on Dataset 2 and Dataset 3 in terms of Accuracy with a 95% Confidence Interval (CI) and Run-time in seconds

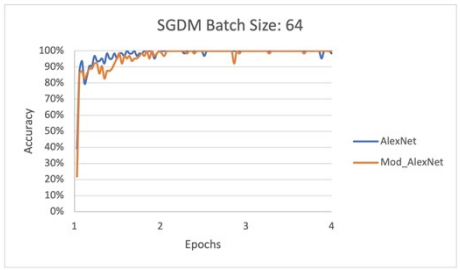
Deep Learning Model	Dataset 2		Dataset 3	
	Accuracy (± 95% CI)	Run-time (sec)	Accuracy (± 95% CI)	Run-time (sec)
ResNet-18	67.43% (±5.30%)	1,535	87.91% (±2.61%)	7,614
AlexNet	77.80% (±4.70%)	1,040	89.60% (±2.44%)	2,562
GoogleNet	63.08% (±5.46%)	1,986	83.47% (±2.97%)	14,874
VGG-16	68.89% (±5.24%)	9,174	88.00% (±2.60%)	22,381
MobileNetV2	68.42% (±5.26%)	1,663	84.47% (±2.90%)	10,120
DenseNet-201	65.40% (±5.38%)	9,259	84.81% (±2.87%)	37,011



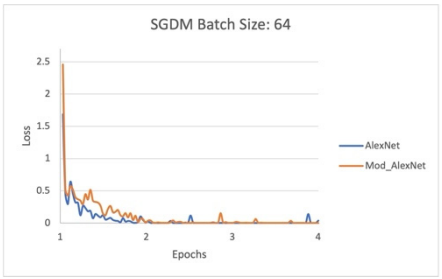
(**a**)



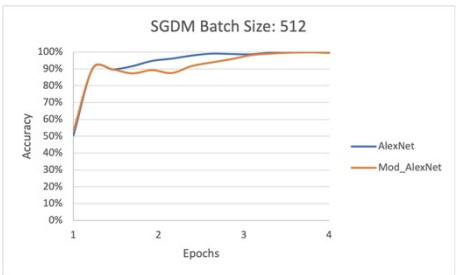
(**b**)



(c)



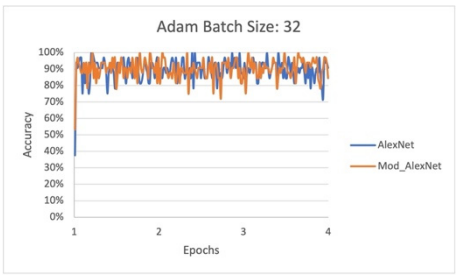
(d)



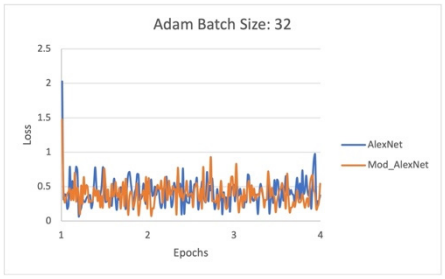
(e)

(f)

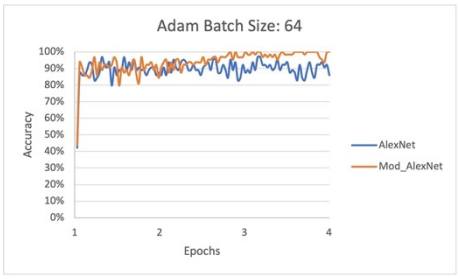
Figure S1. For SGDM optimizer (a) Training Accuracy on Batch size: 32, (b) Training Loss on Batch size: 32, (c) Training Accuracy on Batch size: 64, (d) Training Loss on Batch size: 64, (e) Training Accuracy on Batch size: 512, (f) Training Loss on Batch size: 512.



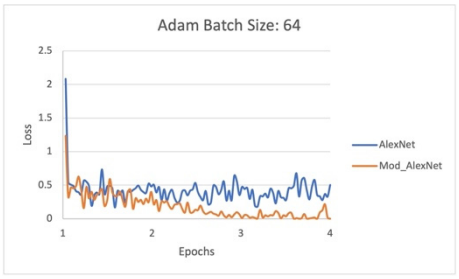
(a)



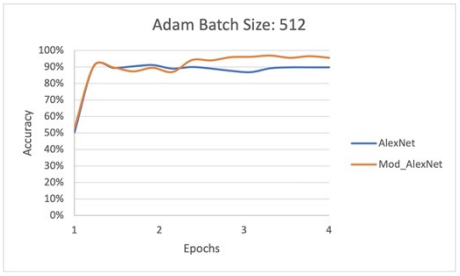
(b)



(c)



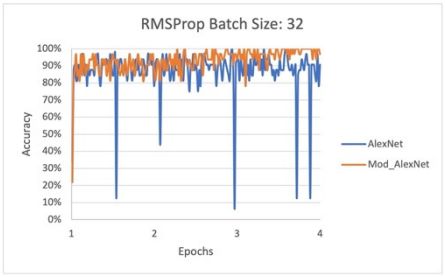
(d)



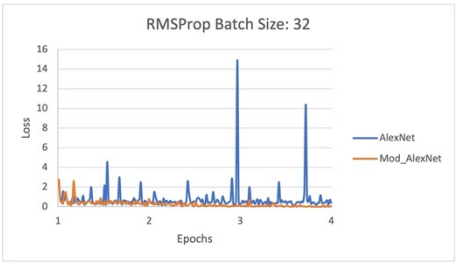
(e)

(f)

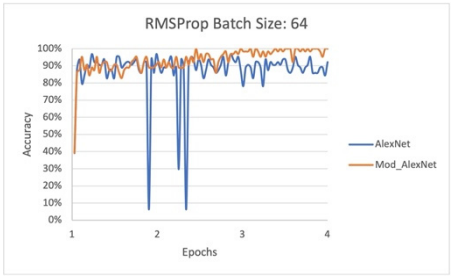
Figure S2. For Adam Optimizer (a) Training Accuracy on Batch size: 32, (b) Training Loss on Batch size: 32, (c) Training Accuracy on Batch size: 64, (d) Training Loss on Batch size: 64, (e) Training Accuracy on Batch size: 512, (f) Training Loss on Batch size: 512.



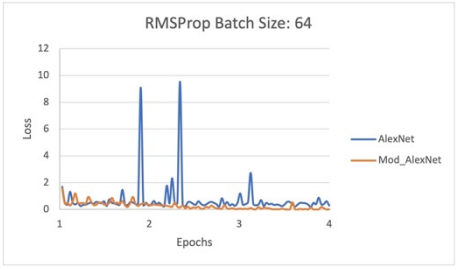
(**a**)



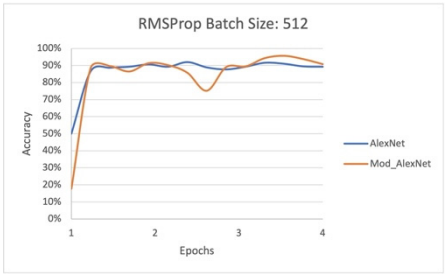
(**b**)



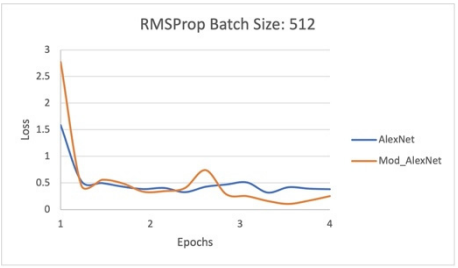
(c)



(d)



(e)



(f)

Figure S3. For RMSProp (a) Training Accuracy on Batch size: 32, (b) Training Loss on Batch size: 32, (c) Training Accuracy on Batch size: 64, (d) Training Loss on Batch size: 64, (e) Training Accuracy on Batch size: 512, (f) Training Loss on Batch size: 512.