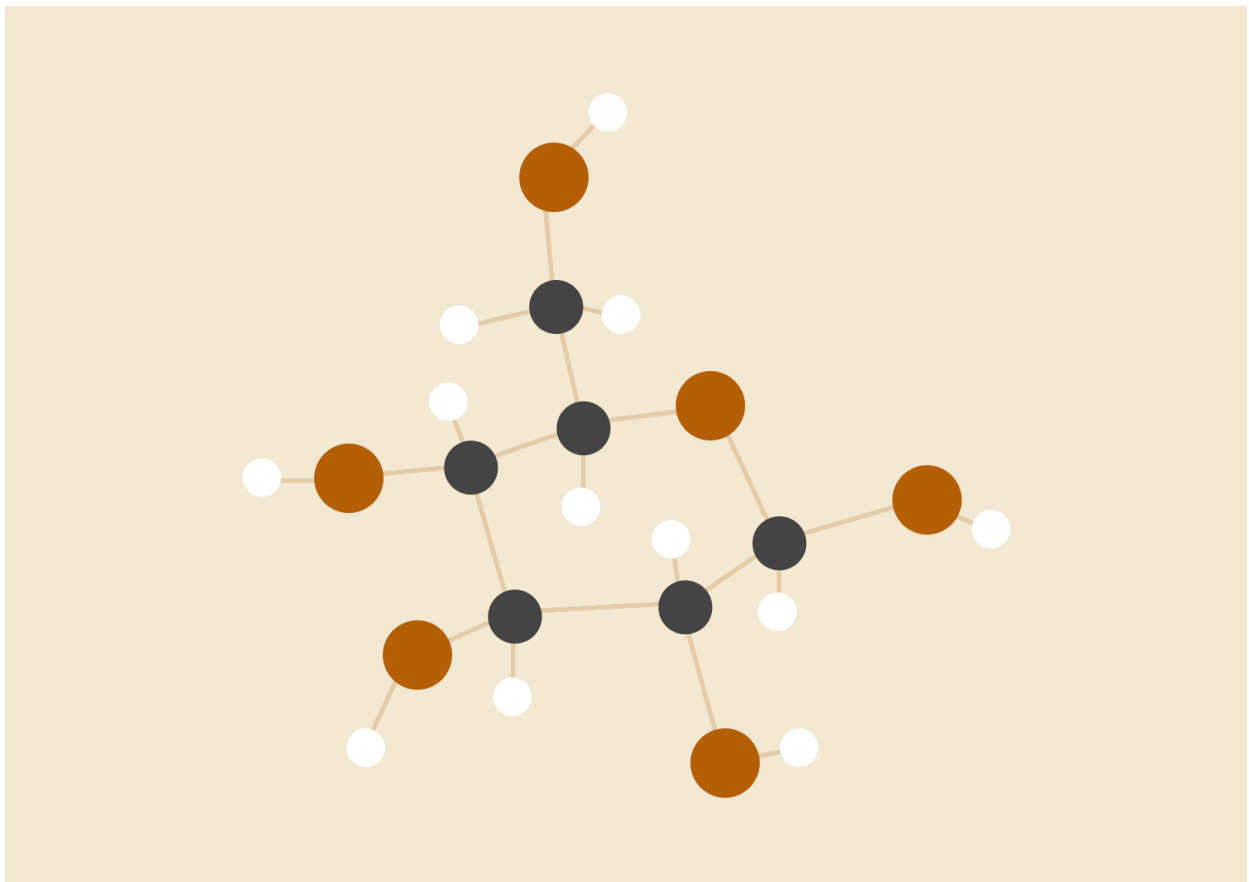


# SISTEME DE OPERARE DISTRIBUITE

*Studiul experimental al performanței programului*

*Tema 1*



**Mutu Gheorghică**

10.11.2022

SAI I

## INTRODUCERE

Studiu experimental: studiem eficiența (i.e., performanța exprimată prin prisma timpului de execuție necesar pentru calculul fiecărei operații grafice), pe baza diversilor factori ce influențează timpul de execuție, a programului paralel vs. cel secvențial.

## PROCEDURA

Variația timpului de execuție a calculului pentru fiecare dintre operațiile grafice implementate, în funcție de următorii parametri:

- dimensiunile  $n \times m$  (i.e. rezoluția) imaginii input (recomandare: folosiți ca sample-uri imagini cu rezoluție mare, i.e. zeci de megapixeli);
- numărul de procese alocat pentru execuția job-ului MPI (respectiv, de observat limitarea dată de numărul de "procesoare"/core-uri disponibile pe cele două calculatoare ce formează clusterul configurat pe care veți testa programul paralel);
- modul de alegere a "divizării" matricii imaginii inițiale în submatrici: divizare 1D (i.e., în "felii" orizontale sau verticale) sau divizare 2D Show / Hide the details
- respectiv modul de trimitere a submatricilor spre prelucrare de către procesele worker: load-balancing static sau dinamic.

Alți factori/parametri ce ar putea influența timpul de execuție:

- caracteristicile hardware ale mașinilor clusterului pe care se rulează programul paralel (i.e., numărul de core-uri HW, cu HT sau nu; rulat direct pe mașina gazdă sau într-o mașină virtuală, etc.);
- gestiunea memoriei virtuale (i.e. rata erorilor de pagină);
- containere Docker (și orchestratoare pentru acestea) pentru un control mai fin al resurselor gazdei alocate pentru execuția programului;

## PARAMETRI

PARAMETRU	VALORI POSIBILE
Metoda de execuție	Secvențială Paralelă
Dimensiuni matrice	n x m
Număr de procese	1-12
Tipul divizării	1-2D
Load-Balancing	Static Dinamic
Communication type	Scatter/Gather Send/Receive

## INPUT

Fișier	Pixels	Coloane	Rânduri	Canale
example01.jpg	47619036	3254	4878	3
example02.jpg	136323072	5504	8256	3
example03.jpg	96238398	5494	5839	3
example04.jpg	218131305	10315	7049	3

## REZULTATE

Toate rezultatele sunt calculate pe binare optimizate cu -O3.

Acțiunile posibile sunt:

- 0 - `ACTION_COLOR_INVERSION`
- 1 - `ACTION_BLUE_AND_RED_SWITCH`
- 2 - `ACTION_GAUSSIAN_BLUR`
- 3 - `ACTION_VERTICAL_FLIP`

Pentru 0, 1 avem Scatter & Gather. Pentru 2, 3 avem Send & Receive.

### WSL2 instanță specificații

Spec	Value
Architecture	x86_64
CPU op-mode(s)	32-bit, 64-bit
Byte Order	Little Endian
Address sizes	36 bits physical, 48 bits virtual
CPU(s)	12
On-line CPU(s) list	0-11
Thread(s) per core	2

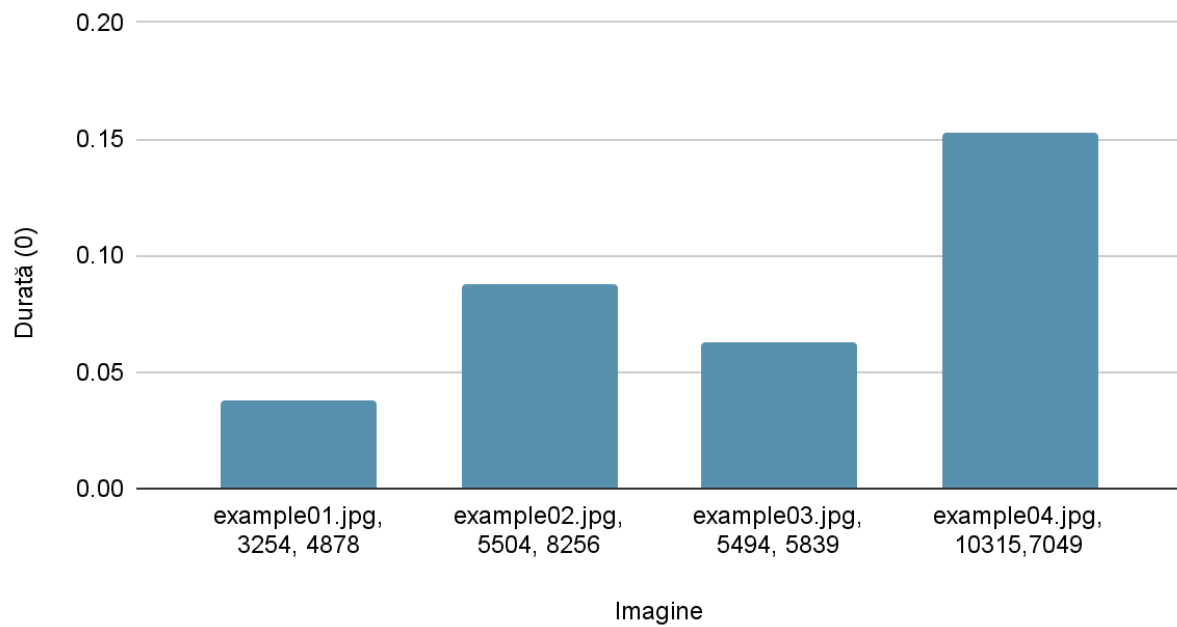
Core(s) per socket	6
Socket(s)	1
Vendor ID	GenuineIntel
CPU family	6
Model	165
Model name	Intel(R) Core(TM) i7-10850H CPU @ 2.70GHz
Stepping	2
CPU MHz	2712.000
CPU max MHz	2712.000
BogoMIPS	5424.00
Hypervisor vendor	Windows Subsystem for Linux
Virtualization type	container
Flags	fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm pni pclmulqdq dtes 64 est tm2 ssse3 fma cx16 xtpr pdcm pcid sse4_1 sse4_2 x2apic movbe popcnt aes xsave osxsave avx f16c rdrand hypervisor lahf_lm abm 3dnowprefetch fsgsbase tsc_adjust b mi1 avx2 smep bmi2 erms invpcid mpx rdseed adx smap clflushopt ibrs ibpb stibp ssbd

## Execuție secvențială (WSL2)

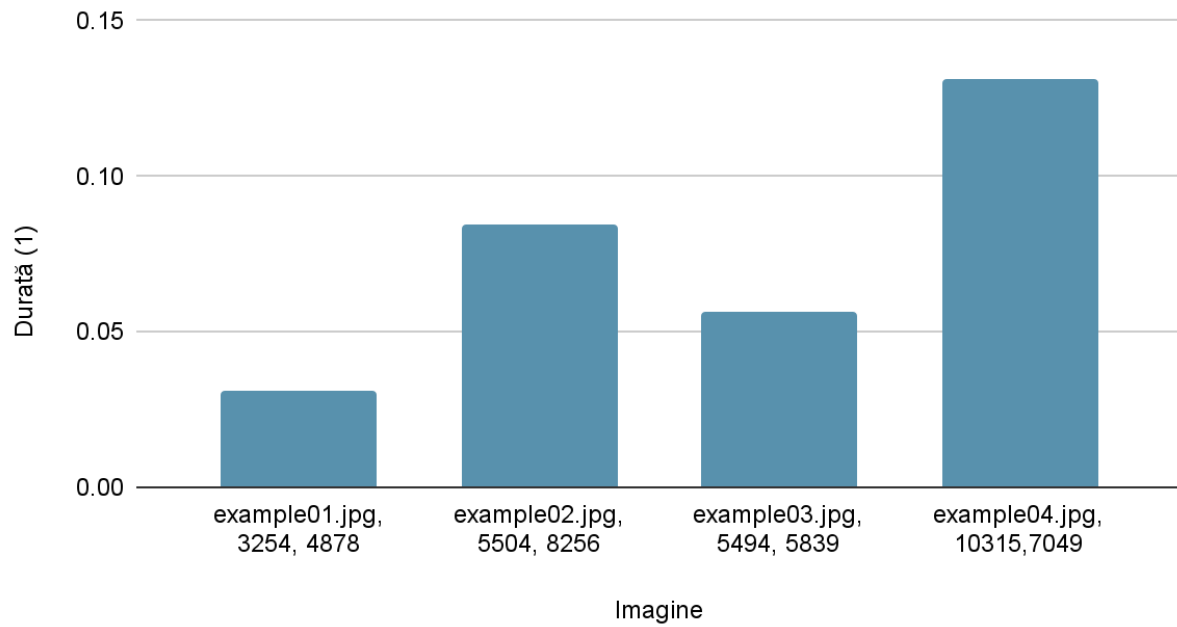
Nume	Randuri	Coloane	#p	Tip execuție	Acțiune	Secunde (media aritmetica a 5 rulări)
example01.jpg	3254	4878	1	secvențială	0	0.037500
example01.jpg	3254	4878	1	secvențială	1	0.031250
example01.jpg	3254	4878	1	secvențială	2 (r5)	4.131250
example01.jpg	3254	4878	1	secvențială	3	0.012500
example02.jpg	5504	8256	1	secvențială	0	0.087500
example02.jpg	5504	8256	1	secvențială	1	0.084375
example02.jpg	5504	8256	1	secvențială	2 (r5)	11.712500
example02.jpg	5504	8256	1	secvențială	3	0.034375
example03.jpg	5494	5830	1	secvențială	0	0.062500
example03.jpg	5494	5830	1	secvențială	1	0.056250
example03.jpg	5494	5830	1	secvențială	2 (r5)	7.543750
example03.jpg	5494	5830	1	secvențială	3	0.025000

example04.jpg	10315	7040	1	secvențială	0	0.153125
example04.jpg	10315	7040	1	secvențială	1	0.131250
example04.jpg	10315	7040	1	secvențială	2 (r5)	17.625000
example04.jpg	10315	7040	1	secvențială	3	0.059375

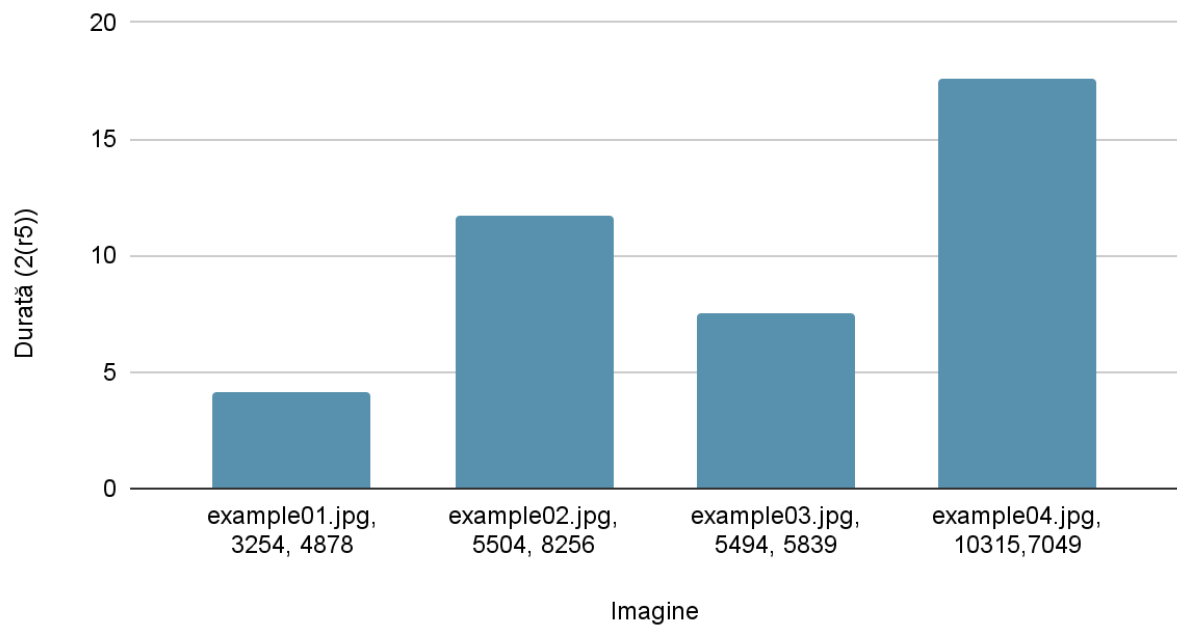
### Durată (0) vs. Imagine



## Durață (1) vs. Imagine

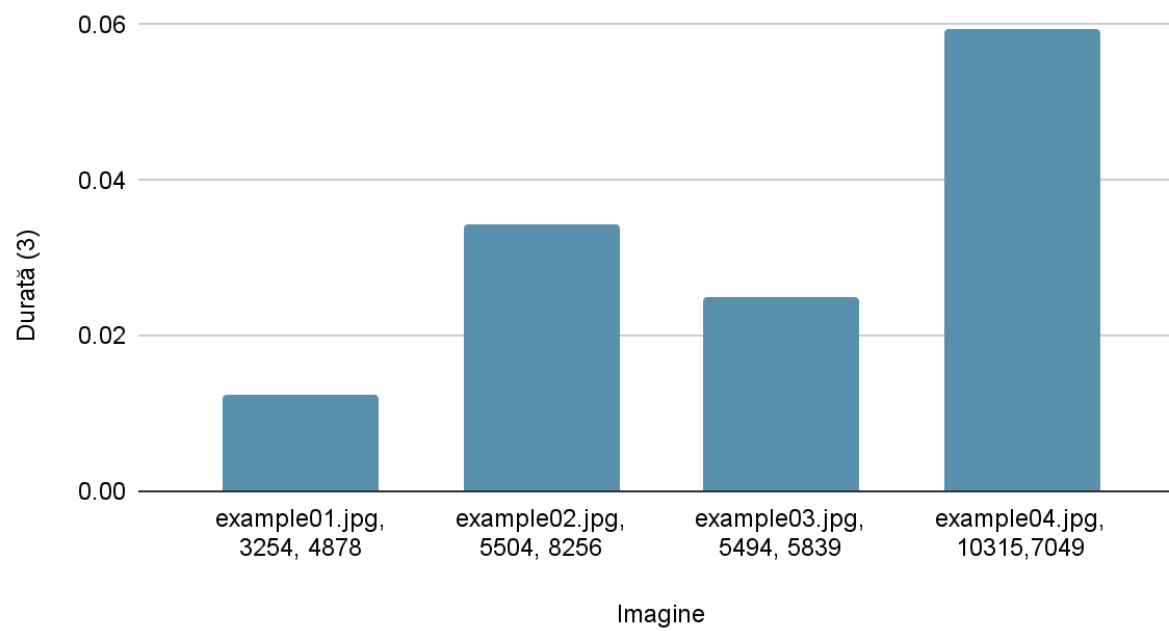


## Durață (2(r5)) vs. Imagine





## Durață (3) vs. Imagine

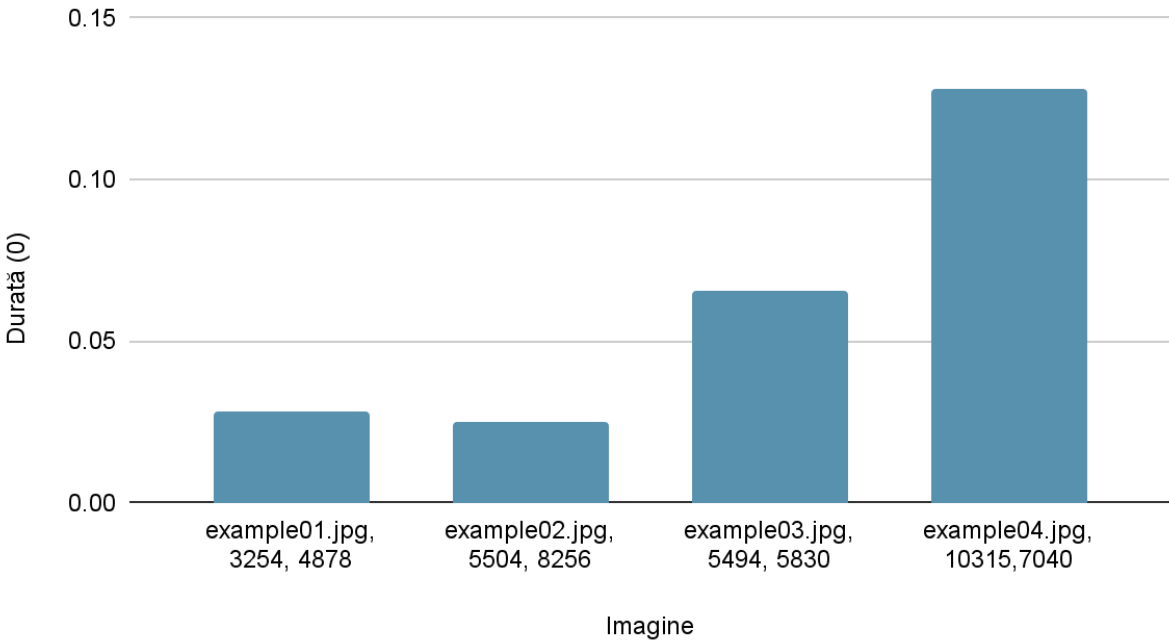


Execuție paralelă (#2) (WSL2) pe aceeași mașină

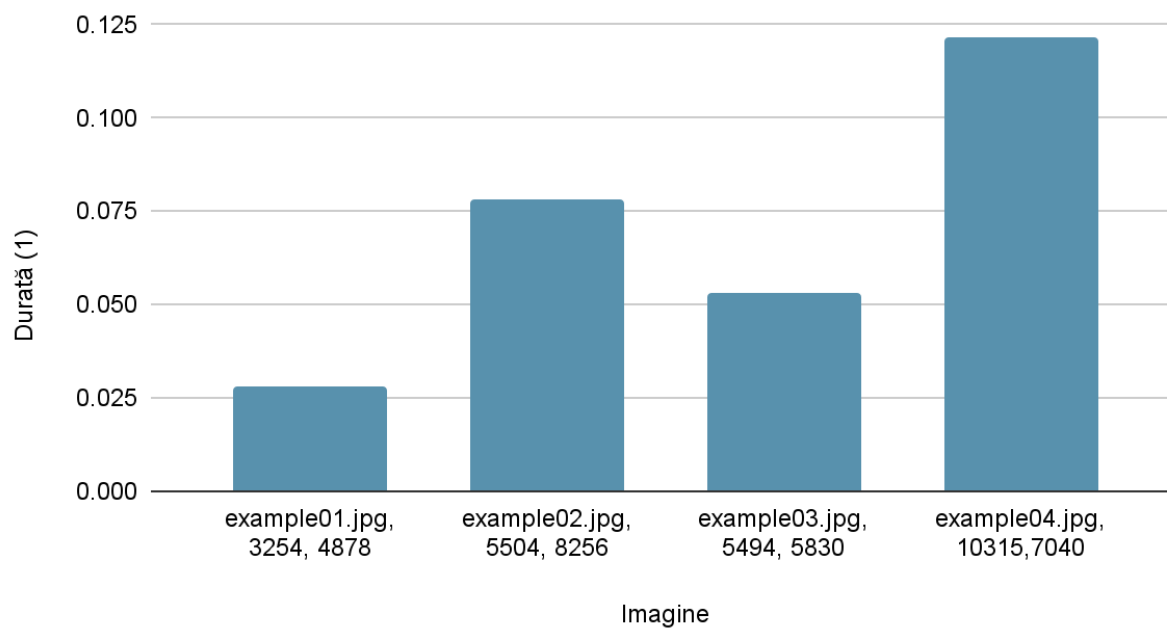
Nume	Randuri	Coloane	#p	Tip execuție	Acțiune	Secunde (media aritmetica a 5 rulări)
example01.jpg	3254	4878	2	paralelă	0	0.028125
example01.jpg	3254	4878	2	paralelă	1	0.025000
example01.jpg	3254	4878	2	paralelă	2 (r5)	0.943750
example01.jpg	3254	4878	2	paralelă	3	0.018750
example02.jpg	5504	8256	2	paralelă	0	0.081250
example02.jpg	5504	8256	2	paralelă	1	0.078125
example02.jpg	5504	8256	2	paralelă	2 (r5)	2.746875
example02.jpg	5504	8256	2	paralelă	3	0.056250
example03.jpg	5494	5830	2	paralelă	0	0.065625
example03.jpg	5494	5830	2	paralelă	1	0.053125
example03.jpg	5494	5830	2	paralelă	2 (r5)	1.987500
example03.jpg	5494	5830	2	paralelă	3	0.040625
example04.jpg	10315	7040	2	paralelă	0	0.128125
example04.jpg	10315	7040	2	paralelă	1	0.121875

example04.jpg	10315	7040	2	paralelă	2 (r5)	4.556250
example04.jpg	10315	7040	2	paralelă	3	0.081250

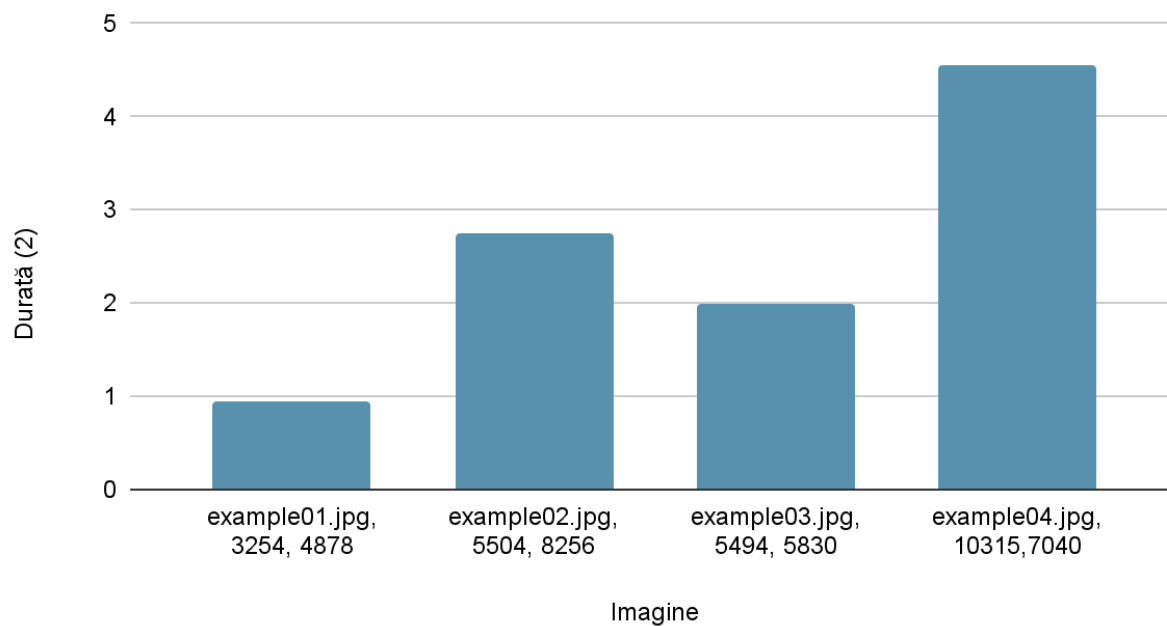
Durață (0) vs. Imagine



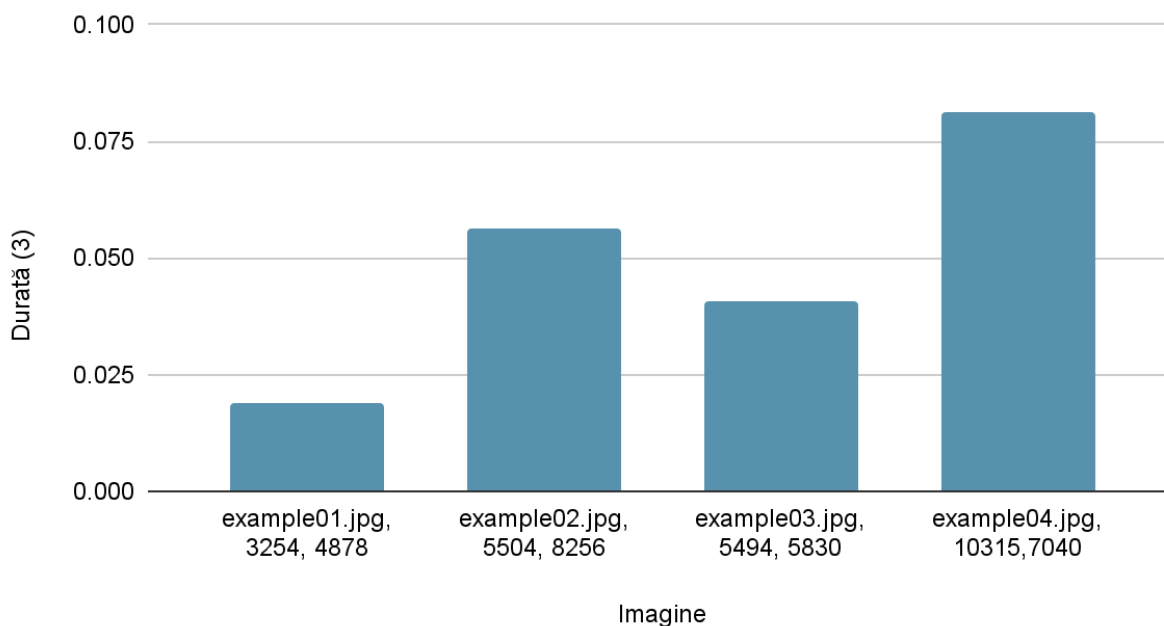
## Durață (1) vs. Imagine



## Durață (2) vs. Imagine



## Durată (3) vs. Imagine



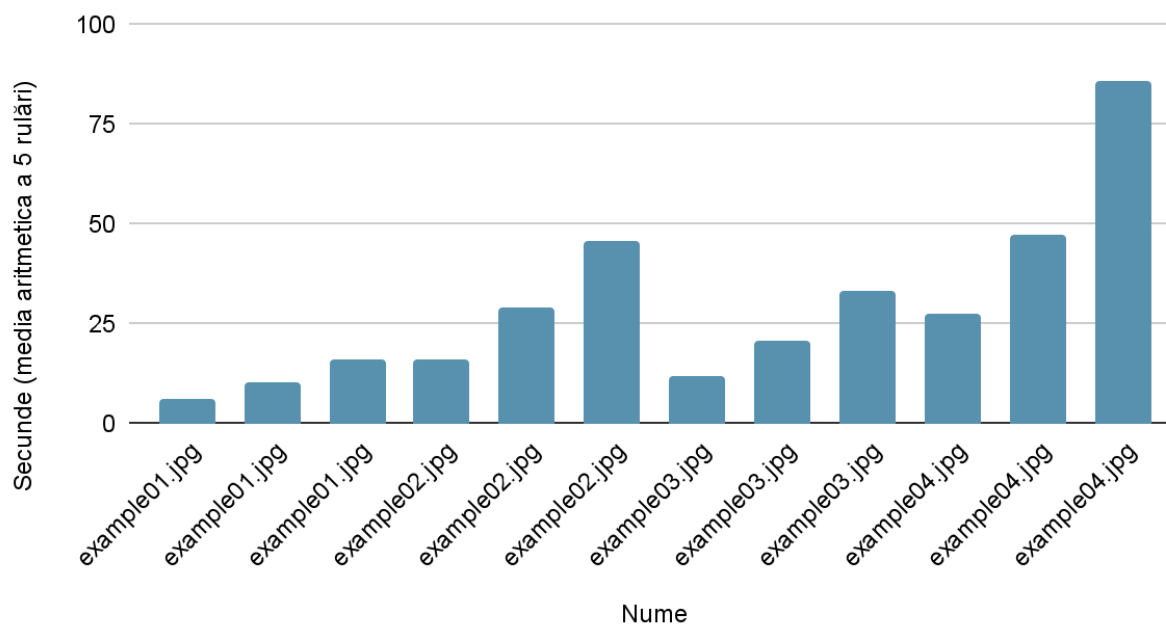
Deja de la 2 procesoare pe WSL începem să avem timpi nesemnificativi pe orice operații în afară de 2 - [ACTION\\_GAUSSIAN\\_BLUR](#). Creștem numărul de procese la 6 dar vom crește și radiusul kernelului la 30, 40 și 50 pentru a avea diferențe semnificative.

Execuție paralelă (#6) (WSL2) pe aceeași mașină

Nume	Randuri	Coloane	#p	Tip execuție	Acțiune	Secunde (media aritmetica a 5 rulări)
example01.jpg	3254	4878	6	paralelă	2 (r30)	6.154688
example01.jpg	3254	4878	6	paralelă	2 (r40)	10.553646
example01.jpg	3254	4878	6	paralelă	2 (r50)	16.309375
example02.jpg	5504	8256	6	paralelă	2 (r30)	16.157813

example02.jpg	5504	8256	6	paralelă	2 (r40)	29.248437
example02.jpg	5504	8256	6	paralelă	2 (r50)	45.809375
example03.jpg	5494	5830	6	paralelă	2 (r30)	11.839583
example03.jpg	5494	5830	6	paralelă	2 (r40)	20.875000
example03.jpg	5494	5830	6	paralelă	2 (r50)	33.435938
example04.jpg	10315	7040	6	paralelă	2 (r30)	27.435937
example04.jpg	10315	7040	6	paralelă	2 (r40)	47.444271
example04.jpg	10315	7040	6	paralelă	2 (r50)	86.056771

Secunde (media aritmetica a 5 rulări) vs. Nume



## Cluster VM specificații

Spec	Value
Architecture	x86_64
CPU op-mode(s)	32-bit, 64-bit
Byte Order	Little Endian
Address sizes	39 bits physical, 48 bits virtual
CPU(s)	2
On-line CPU(s) list	0-1
Thread(s) per core	1
Core(s) per socket	1
Socket(s)	1
Vendor ID	GenuineIntel
CPU family	6
Model	165
Model name	Intel(R) Core(TM) i7-10850H CPU @ 2.70GHz
Stepping	2
CPU MHz	2711.999

CPU max MHz	2712.000
BogoMIPS	5423.99
Hypervisor vendor	Microsoft
Virtualization type	full
L1d cache	32 KiB
L1i cache	32 KiB
L2 cache	256 KiB
L3 cache	12 MiB
NUMA node0 CPU(s)	0,1
Vulnerability Itlb multihit	KVM: Vulnerable
Vulnerability L1tf	Not affected
Vulnerability Mds	Not affected
Vulnerability Meltdown	Not affected
Vulnerability Spec store bypass	Mitigation; Speculative Store Bypass disabled via prctl and seccomp
Vulnerability Spectre v1	Mitigation; usercopy/swapgs barriers and __user pointer sanitization
Vulnerability Spectre v2	Mitigation; Enhanced IBRS, IBPB conditional, RSB filling
Vulnerability Srbds	Not affected



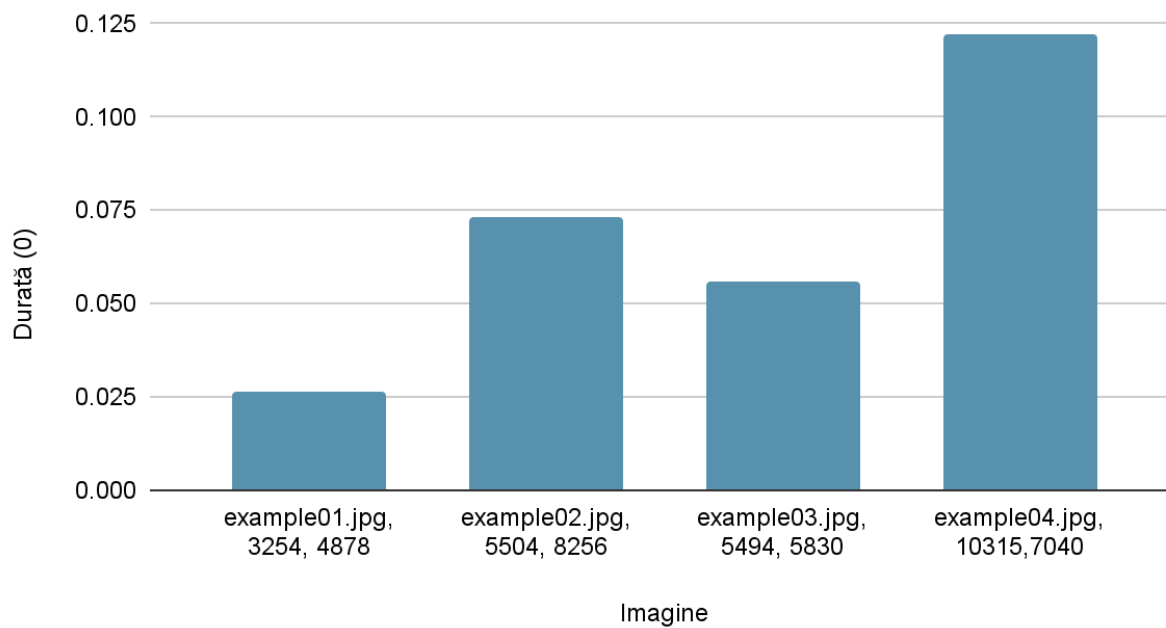
Vulnerability Tsx async abort	Not affected
Flags	fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ss ht syscall nx pdpe1gb rdtscp lm constant_tsc rep_good nopl xtopology cpuid pni pclmulqdq ssse3 fma cx16 pcid sse_4_1 sse4_2 movbe popcnt aes xsave avx f16c rdrand hypervisor lahf_lm abm 3dnowprefetch invpcid_single ssbd ibrs ibpb stibp ibrs_enhanced fsgsbase bmi1 avx2 smep bmi2 erms invpcid rdseed adx smap clflushopt xsaveopt xsaves xgetbv1 xsaves flush_l1d arch_capabilities

#### Execuție secvențială (cluster VM)

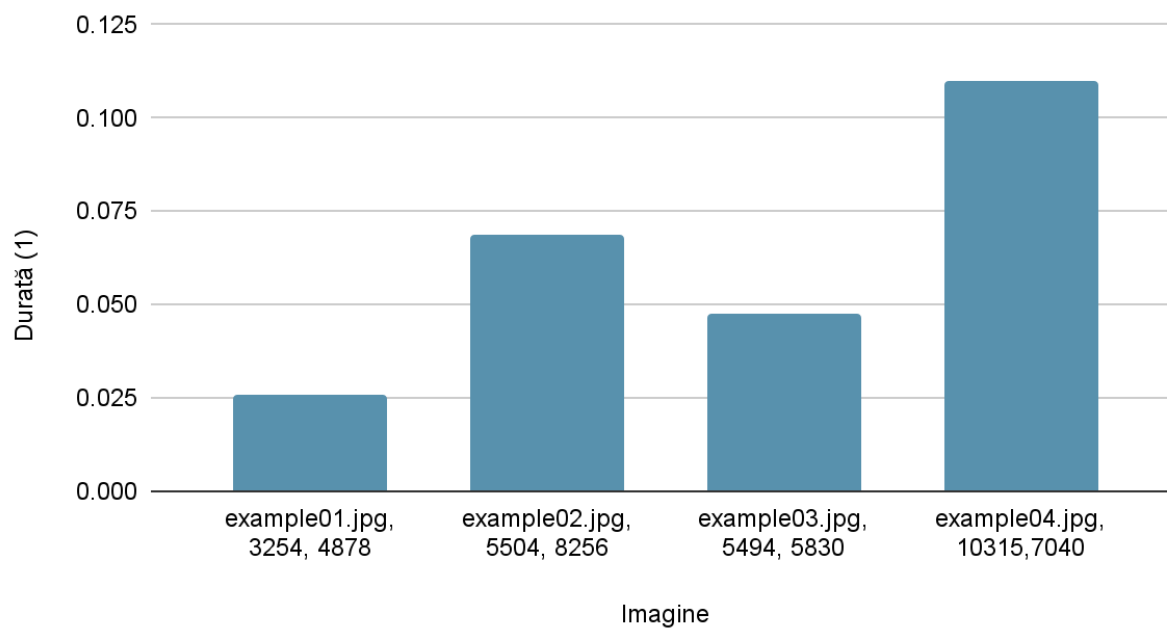
Nume	Randuri	Coloane	#p	Tip execuție	Acțiuni	Secunde (media aritmetica a 5 rulări)
example01.jpg	3254	4878	1	secvențială	0	0.026578
example01.jpg	3254	4878	1	secvențială	1	0.025662
example01.jpg	3254	4878	1	secvențială	2 (r5)	3.622666
example01.jpg	3254	4878	1	secvențială	3	0.011744
example02.jpg	5504	8256	1	secvențială	0	0.073243
example02.jpg	5504	8256	1	secvențială	1	0.068831
example02.jpg	5504	8256	1	secvențială	2 (r5)	10.456979
example02.jpg	5504	8256	1	secvențială	3	0.031446
example03.jpg	5494	5830	1	secvențială	0	0.056116

example03.jpg	5494	5830	1	secvențială	1	0.047791
example03.jpg	5494	5830	1	secvențială	2 (r5)	7.369922
example03.jpg	5494	5830	1	secvențială	3	0.022672
example04.jpg	10315	7040	1	secvențială	0	0.122126
example04.jpg	10315	7040	1	secvențială	1	0.109964
example04.jpg	10315	7040	1	secvențială	2 (r5)	17.521252
example04.jpg	10315	7040	1	secvențială	3	0.048693

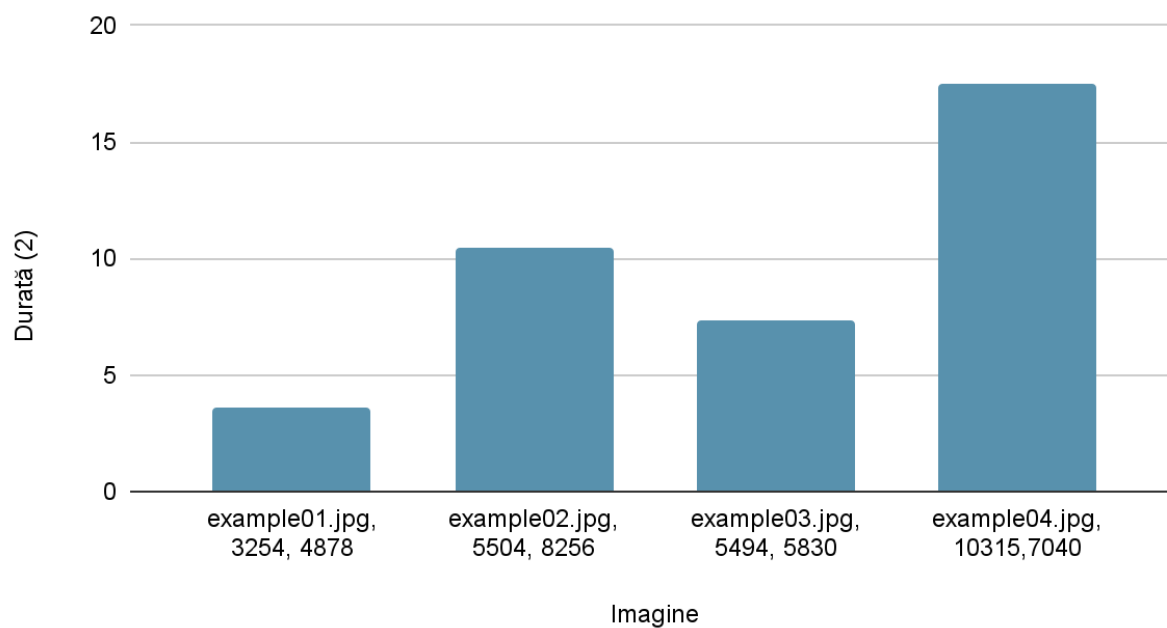
### Durață (0) vs. Imagine



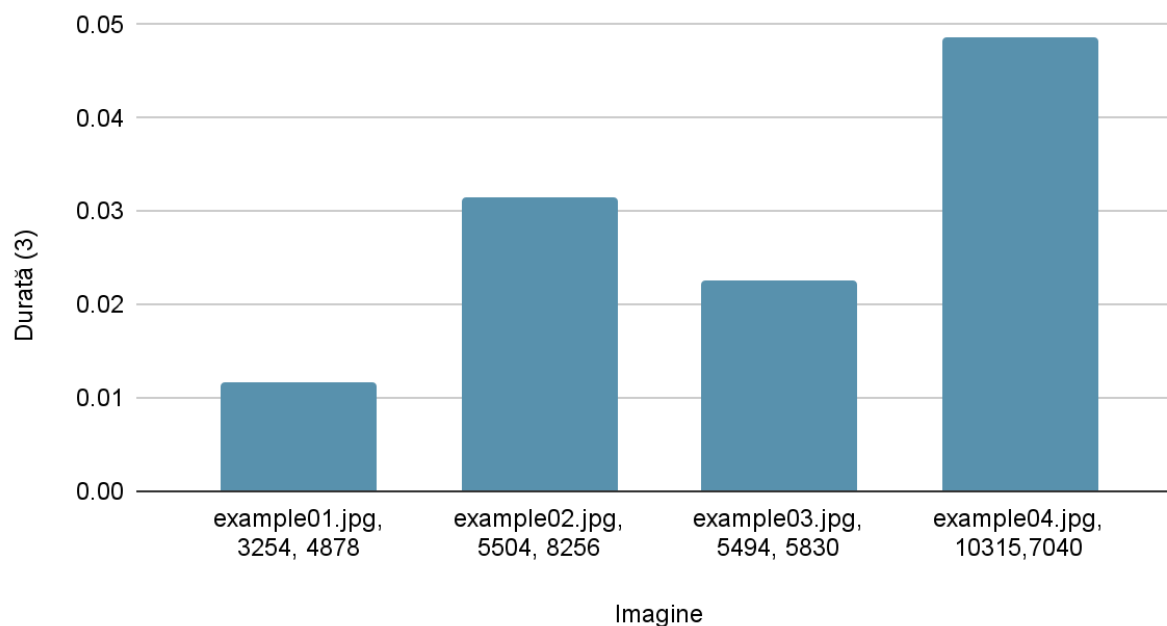
## Durață (1) vs. Imagine



## Durață (2) vs. Imagine



## Durată (3) vs. Imagine

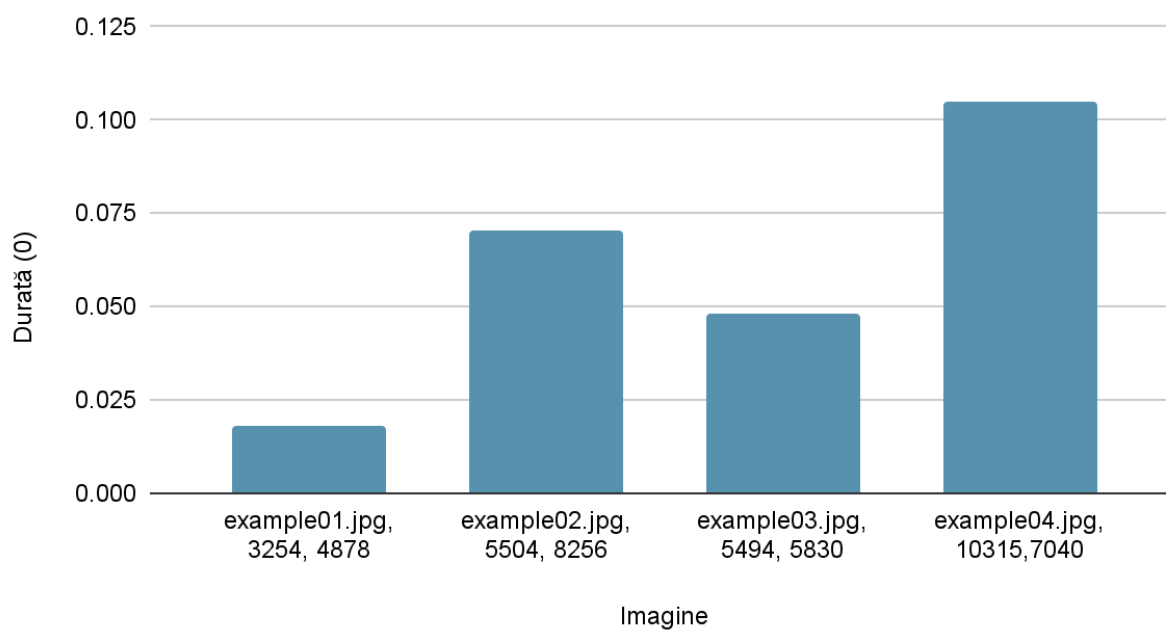


## Execuție paralelă (#2) (cluster VM) pe aceeași mașină

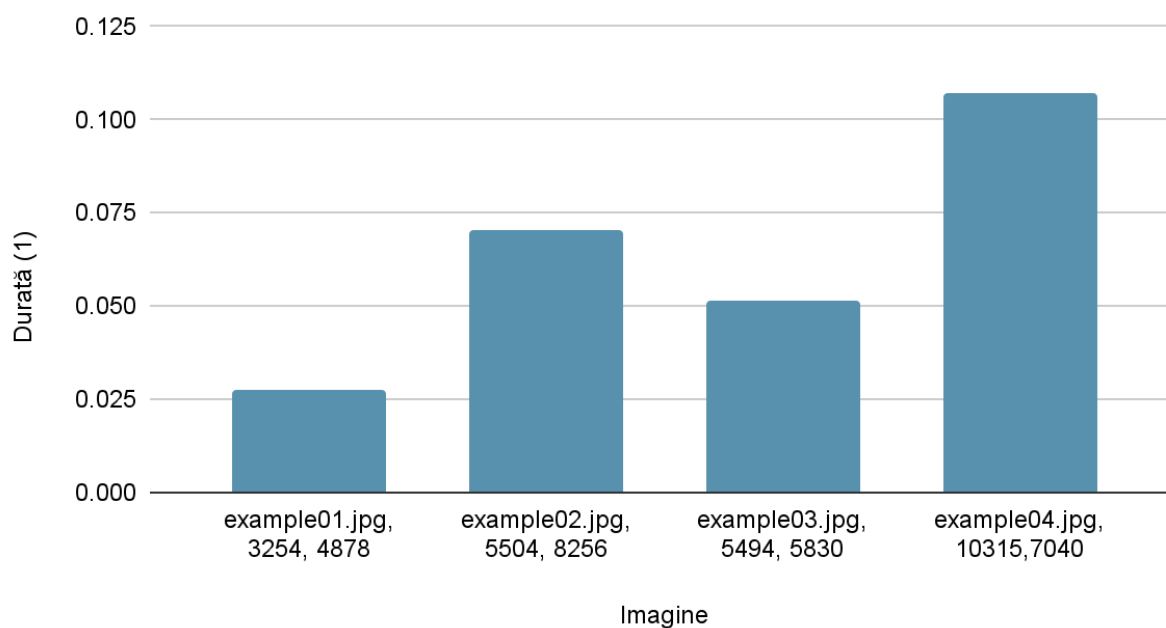
Nume	Randuri	Coloane	#p	Tip execuție	Acțiuni	Secunde (media aritmetica a 5 rulări)
example01.jpg	3254	4878	2	paralelă	0	0.017986
example01.jpg	3254	4878	2	paralelă	1	0.027423
example01.jpg	3254	4878	2	paralelă	2 (r5)	1.041350
example01.jpg	3254	4878	2	paralelă	3	0.021421
example02.jpg	5504	8256	2	paralelă	0	0.070463
example02.jpg	5504	8256	2	paralelă	1	0.070134

example02.jpg	5504	8256	2	paralelă	2 (r5)	3.143462
example02.jpg	5504	8256	2	paralelă	3	0.052785
example03.jpg	5494	5830	2	paralelă	0	0.047854
example03.jpg	5494	5830	2	paralelă	1	0.051390
example03.jpg	5494	5830	2	paralelă	2 (r5)	1.987500
example03.jpg	5494	5830	2	paralelă	3	0.037593
example04.jpg	10315	7040	2	paralelă	0	0.104874
example04.jpg	10315	7040	2	paralelă	1	0.107259
example04.jpg	10315	7040	2	paralelă	2 (r5)	2.177675
example04.jpg	10315	7040	2	paralelă	3	0.068907

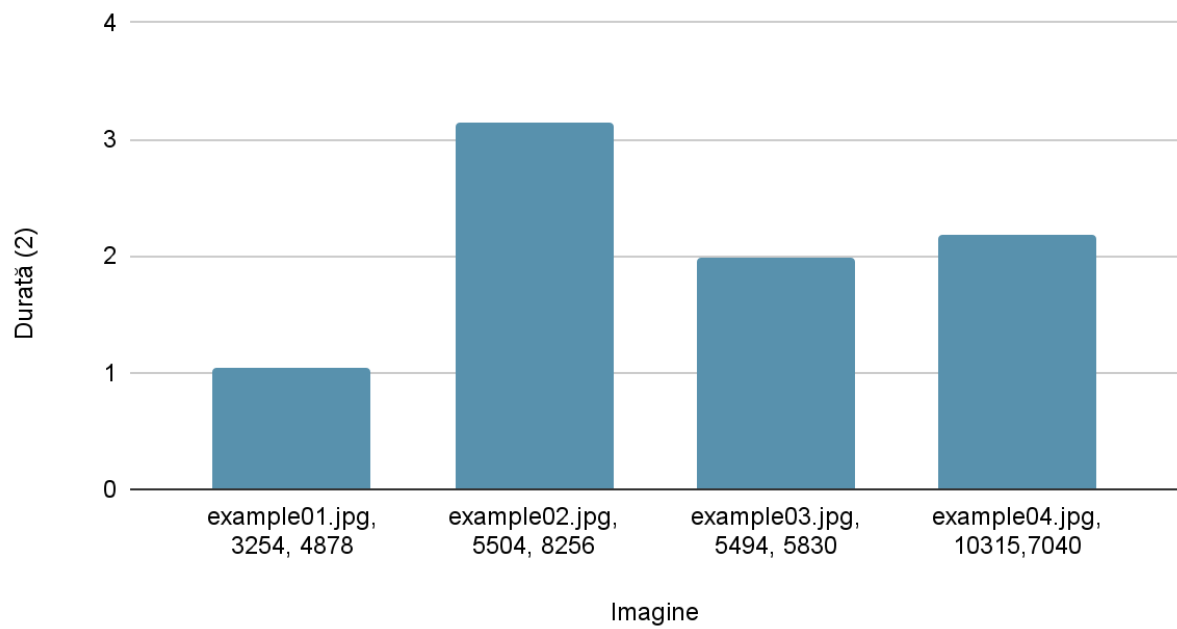
## Durață (0) vs. Imagine



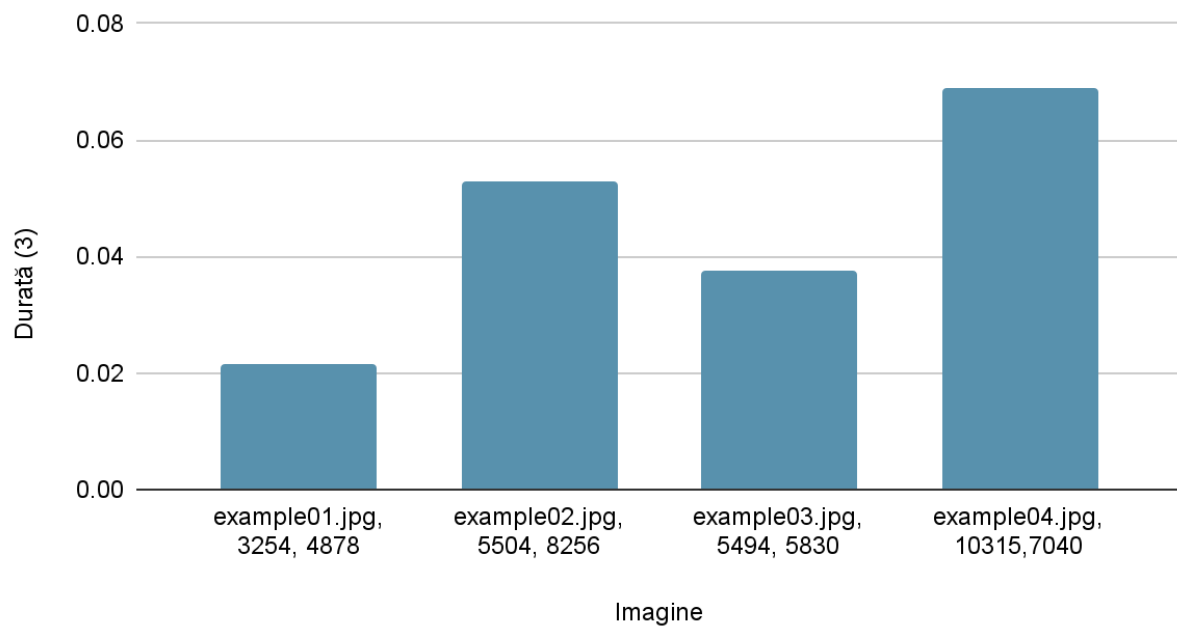
## Durață (1) vs. Imagine



Durață (2) vs. Imagine



Durață (3) vs. Imagine



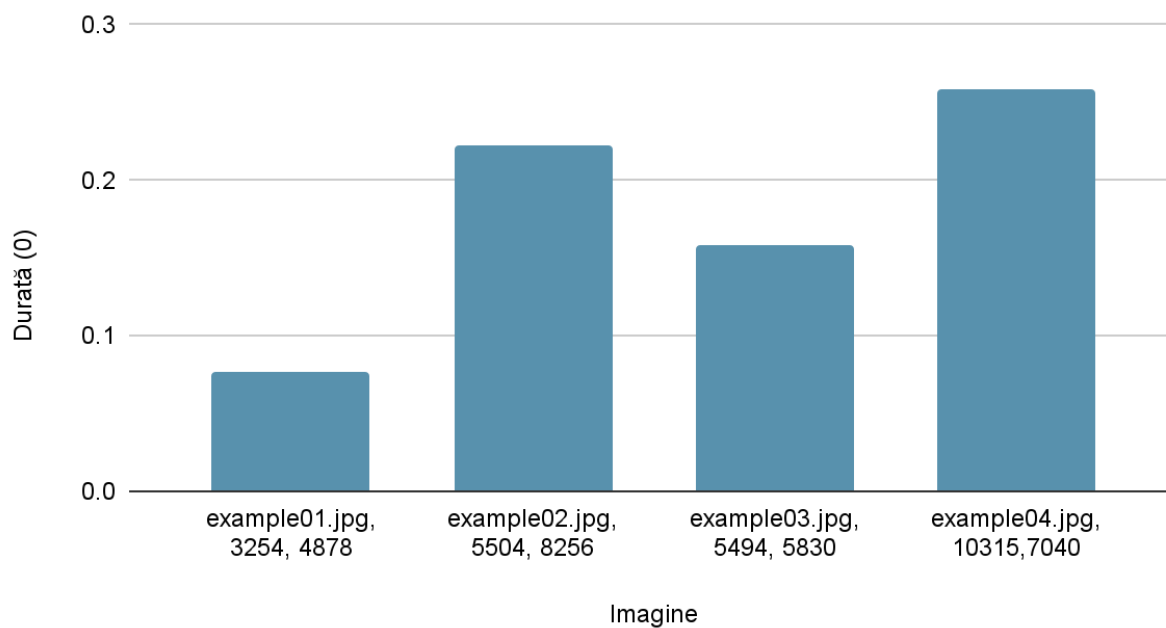
Execuție paralelă (#6) (cluster VM) pe 3 mașini

Nume	Randuri	Coloane	#p	Tip execuție	Acțiune	Secunde (media aritmetica a 5 rulări)
example01.jpg	3254	4878	6	paralelă	0	0.076858
example01.jpg	3254	4878	6	paralelă	1	0.076863
example01.jpg	3254	4878	6	paralelă	2 (r5)	0.289518
example01.jpg	3254	4878	6	paralelă	3	0.082530
example02.jpg	5504	8256	6	paralelă	0	0.222178
example02.jpg	5504	8256	6	paralelă	1	0.188356
example02.jpg	5504	8256	6	paralelă	2 (r5)	0.472114
example02.jpg	5504	8256	6	paralelă	3	0.219584
example03.jpg	5494	5830	6	paralelă	0	0.158759
example03.jpg	5494	5830	6	paralelă	1	0.129069
example03.jpg	5494	5830	6	paralelă	2 (r5)	0.573328
example03.jpg	5494	5830	6	paralelă	3	0.171056
example04.jpg	10315	7040	6	paralelă	0	0.258141
example04.jpg	10315	7040	6	paralelă	1	0.265265

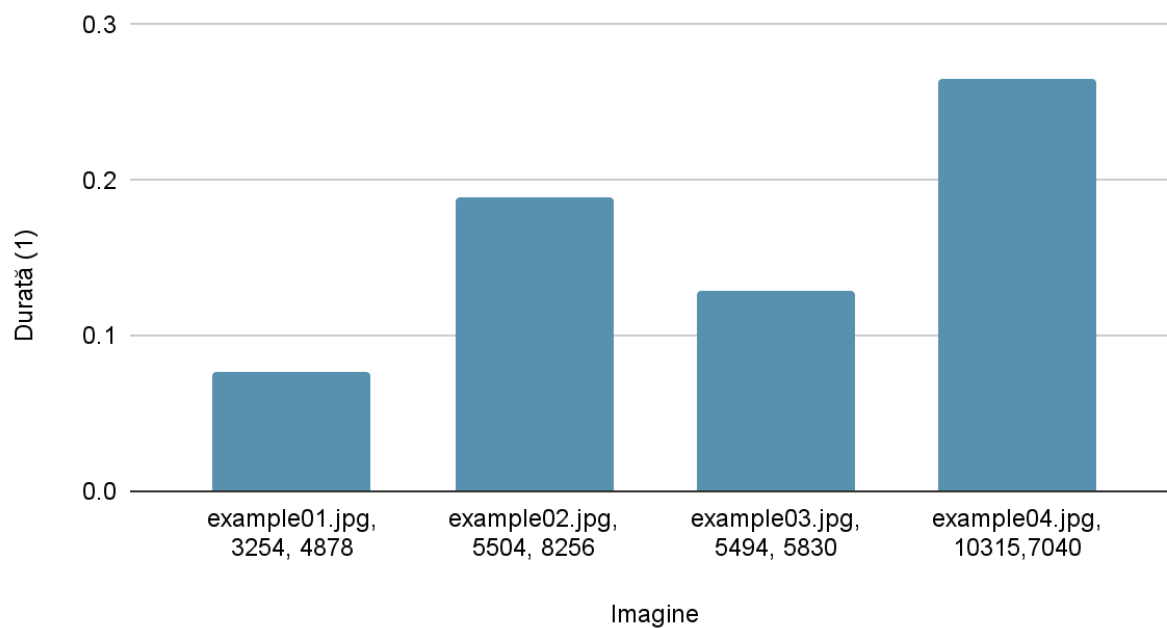


example04.jpg	10315	7040	6	paralelă	2 (r5)	1.291921
example04.jpg	10315	7040	6	paralelă	3	0.359761

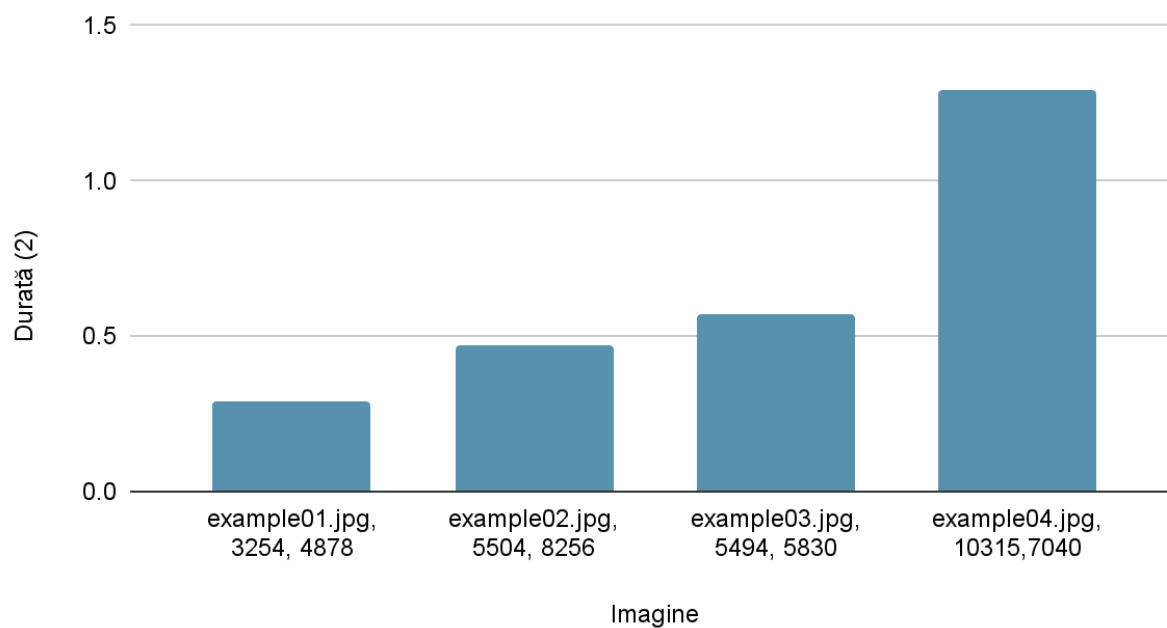
## Durată (0) vs. Imagine



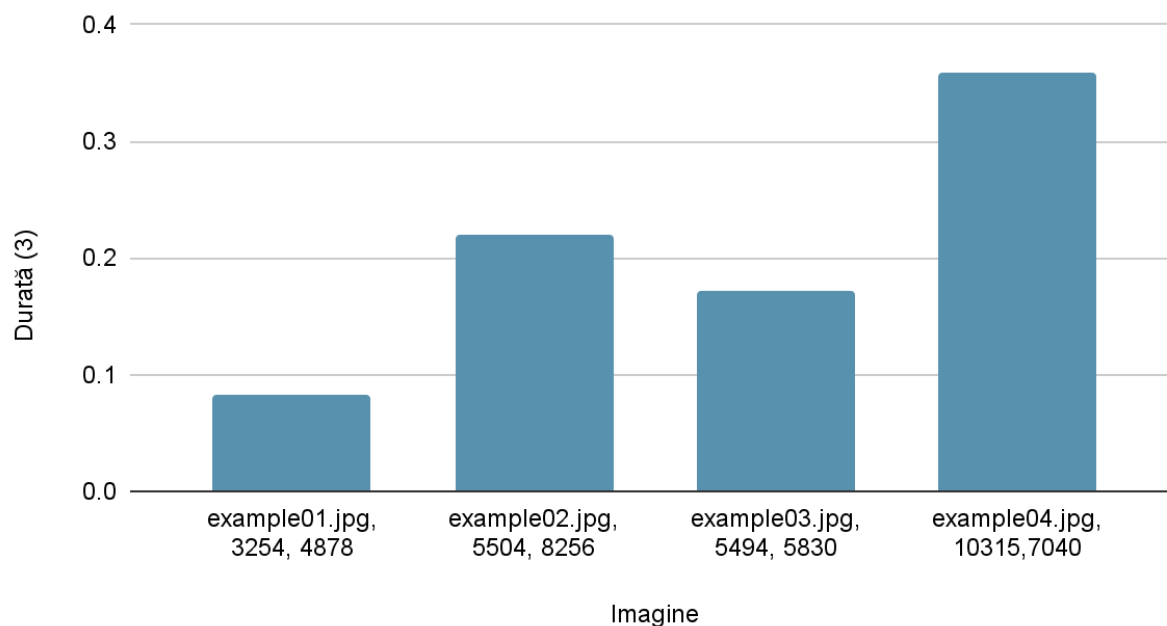
## Durață (1) vs. Imagine



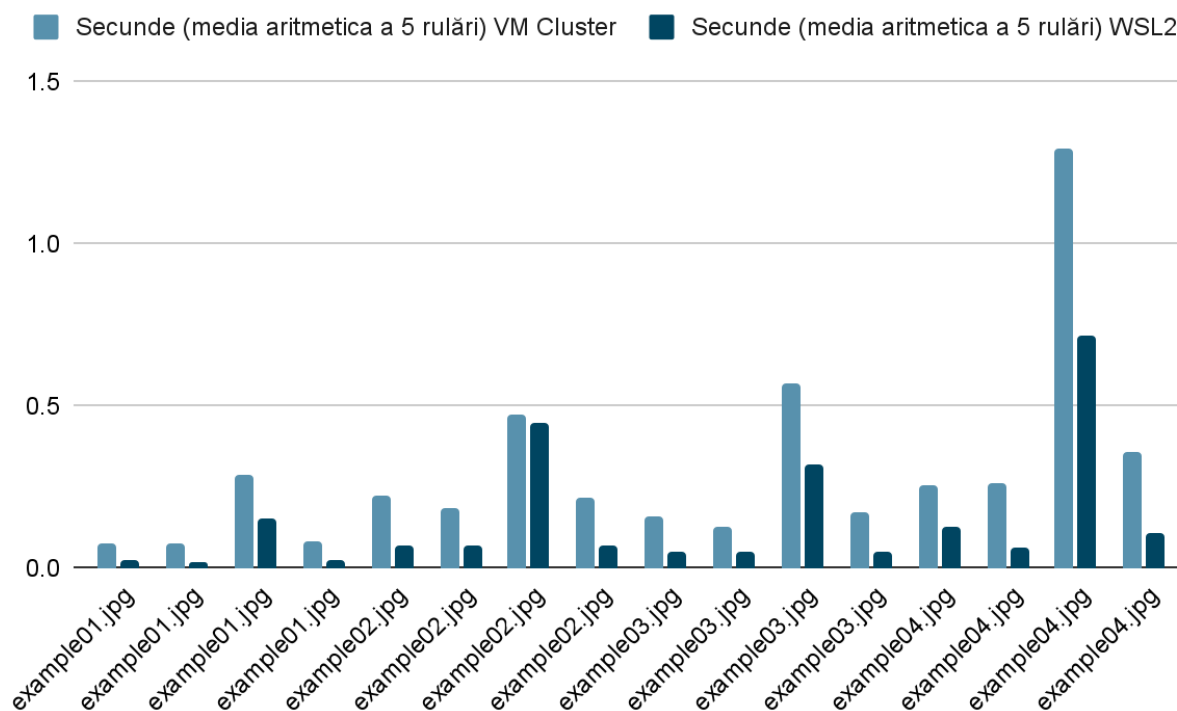
## Durață (2) vs. Imagine



## Durată (3) vs. Imagine



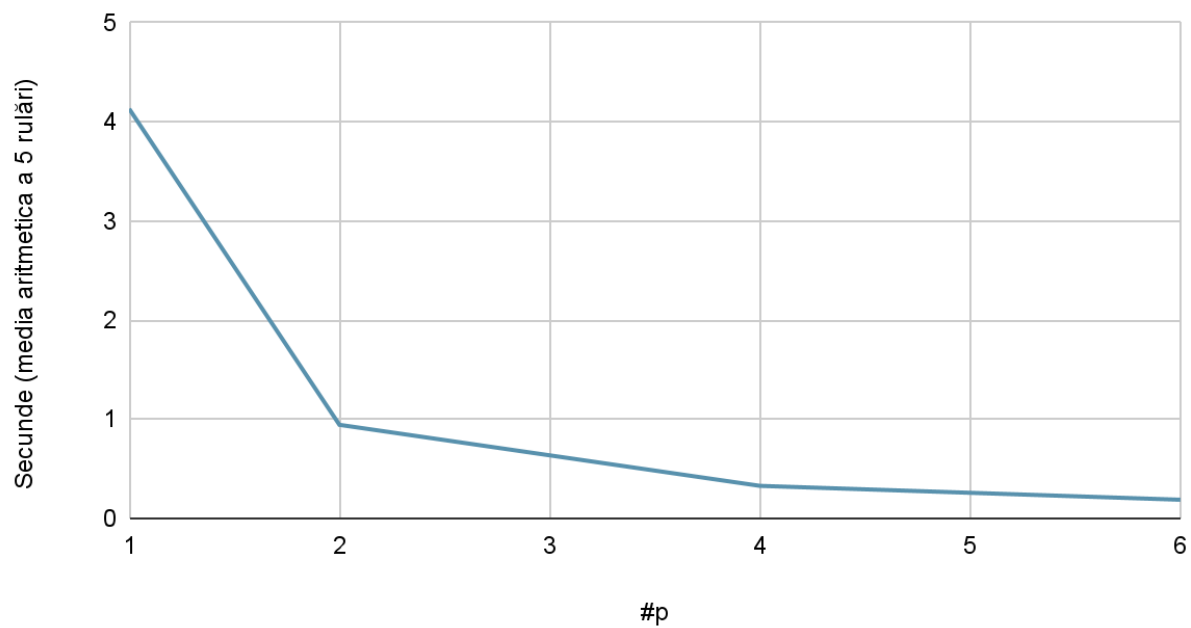
## Execuție paralelă (#6) (WSL2) pe aceeași mașină vs Execuție paralelă (#6) (cluster VM) pe 3 mașini



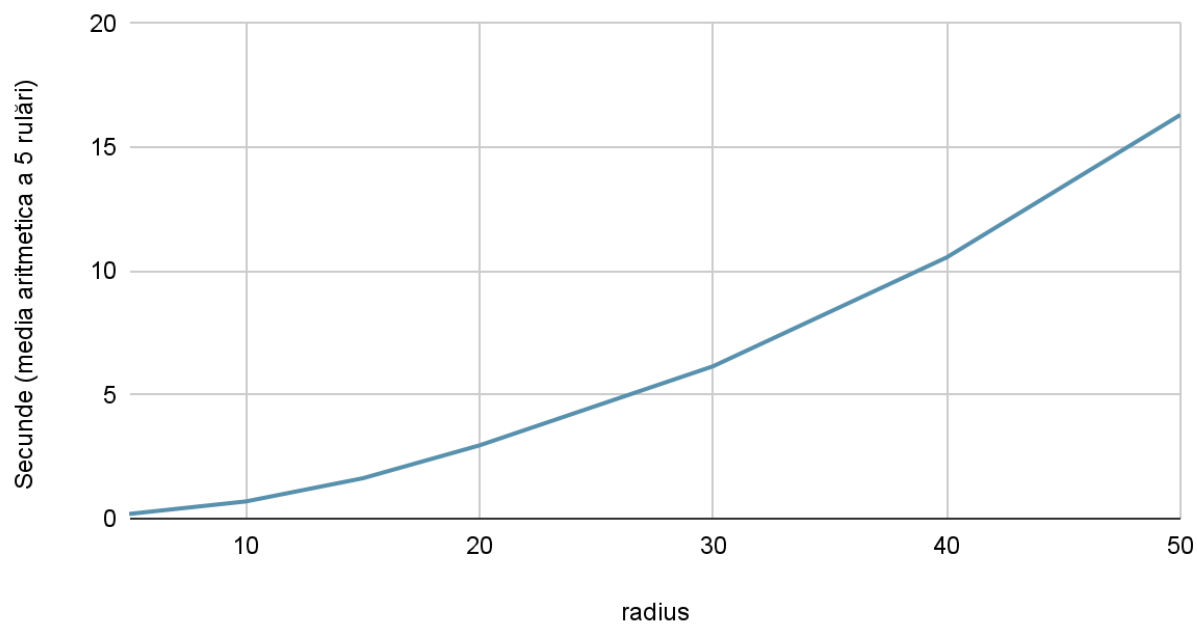
### Evoluția duratei pentru Gaussian Blur (image01)

Nume	Machine	Execuție	#p	radius	Secunde (media aritmetica a 5 rulări)
example01.jpg	WSL2	secvențială	1	5	4.131250
example01.jpg	WSL2	paralelă	2	5	0.943750
example01.jpg	WSL2	paralelă	4	5	0.328125
example01.jpg	WSL2	paralelă	6	5	0.188021
example01.jpg	WSL2	paralelă	6	10	0.693229
example01.jpg	WSL2	paralelă	6	15	1.628646
example01.jpg	WSL2	paralelă	6	20	2.958333
example01.jpg	WSL2	paralelă	6	30	6.154688
example01.jpg	WSL2	paralelă	6	40	10.553646
example01.jpg	WSL2	paralelă	6	50	16.309375
example01.jpg	VM	secvențială	1	5	3.622666
example01.jpg	VM	paralelă	2	5	1.041350
example01.jpg	VM	paralelă	6	5	0.289518

Secunde (media aritmetica a 5 rulări) vs. #p



Secunde (media aritmetica a 5 rulări) vs. radius



## CONCLUZIE

Execuția secvențială în WSL2 este mai înceată decât pe un VM full folosind Hyper-V. Execuția paralelizată pe WSL2 (o singură mașină) este mai rapidă decât execuția pe 3 VMuri full folosind Hyper-V.

Costul este simțitor atunci când datele de input sunt multe ca număr (date fiind întârzierile provocate de transferul datelor în rețea (ssh)) deși taskurile sunt procesate mai rapid pe VMuri.

Când taskurile sunt mai mari (matricea sau inputul unui singur file este mult mai mare) atunci merită plătit costul rețelei pentru transferul datelor.

Diferențele duratei de procesare între imaginile care au număr de pixeli diferiți sunt date, în principal, de numărul de rânduri (atât scatter/gather cât și send/receive procesează sliceuri de rânduri).

Împărțirea prin scatter/gather sau send/receive este optimă pentru fiecare caz - dacă dimensiunile matricilor cresc sau radiusul blurului (o altă matrice) se mărește, creșterea duratei de procesare nu este exponențială.