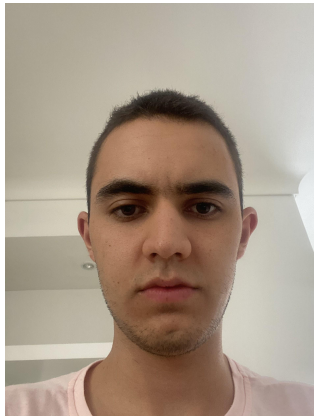
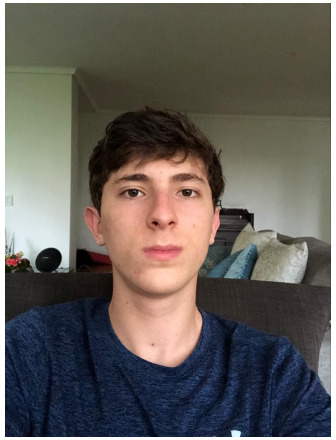
Algorithms to Further the Development of PLA Technologies

UNIVERSIDAD EAFIT

Team Presentation



Tomas Calle



Juan Camilo Salazar

Miguel Angel Cabrera

Simón Marín

Mauricio Toro

https://github.com/tomasCalletce/Algorithms-to-Further-the-Development-of-PLA-technologies

UNIVERSIDAD EAFIT

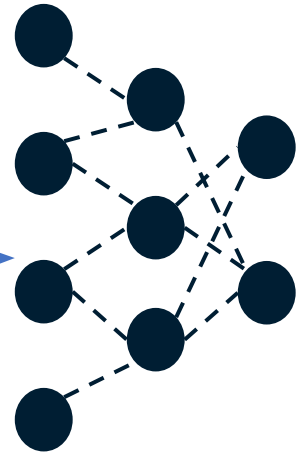
Training Process

Sick-Cattle Images

Healthy-Cattle Images

Convolutional Neural Network

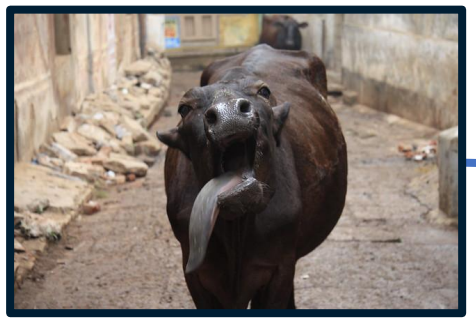
Classification Algorithm



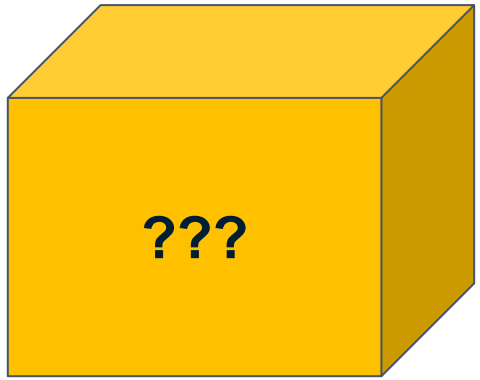
Classification Model

UNIVERSIDAD EAFIT

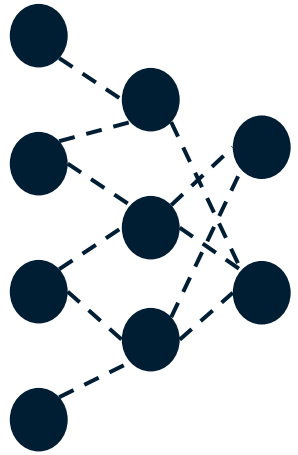
Testing Process



Cattle Image



Interpolation Algorithm



Classification Model

Output

Lossy Compression Algorithm

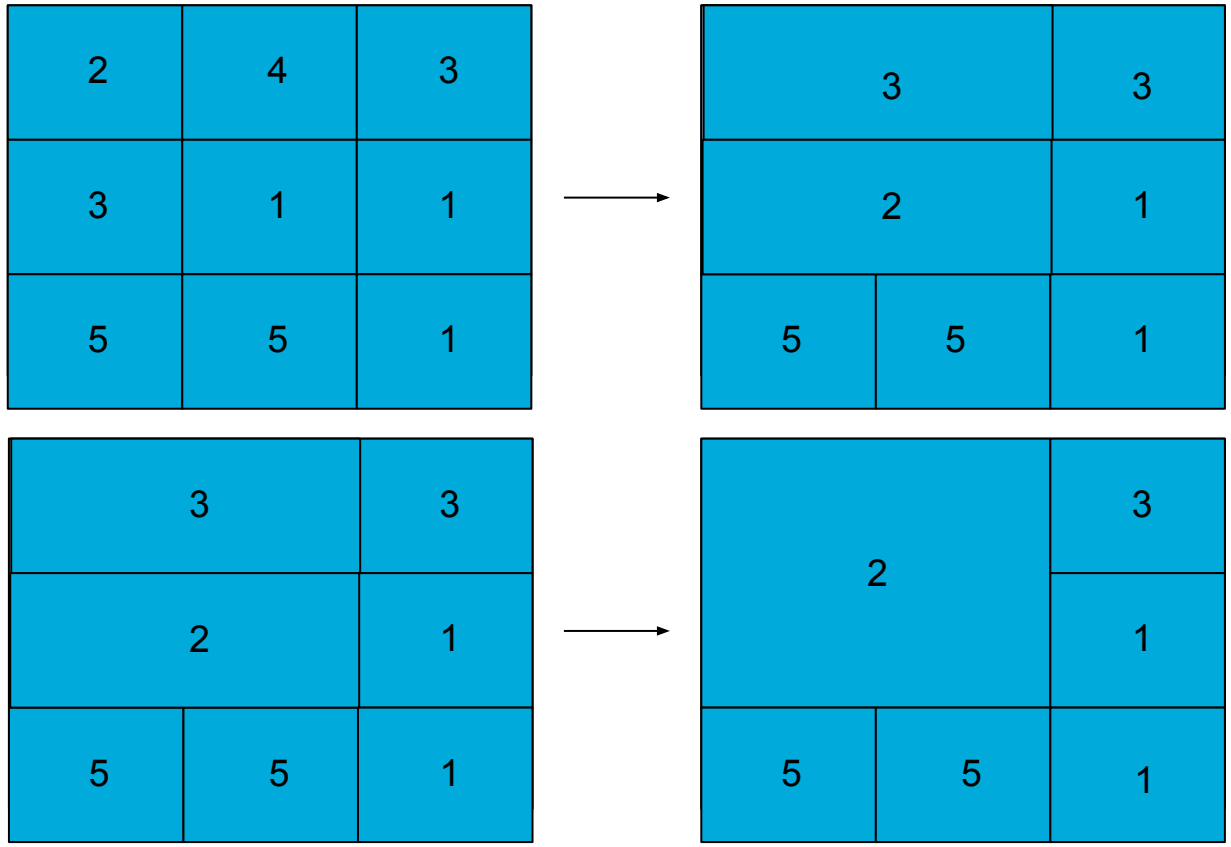


Image compression algorithm - Interpolation Algorithm
The algorithms converts n values into 1 value that represents the average of n values.

Lossy Compression Algorithm

for

2	4
3	1

(2+4)/2 = 3 = x1

(3+1) / 2 = 2 = x2 →

(x1+ x2)/2 = 2

2

The algorithm iterates over blocks of 4 values and calculates the average of all values.

Obstacles LZ77

Test Cases:

back window size: 100
front window size: 1000
# of matches found: 32,368
time: 3.08 s

back window size: 100
front window size: 2000
# of matches found: 42,774
time: 4.4 s

back window size: 100
front window size: 30000
# of matches found: 54,950
time: 6.68

back window size: 4000
front window size: 100
# of matches found: 77,83
time: 7,78

Problems to encounter:

1. Time complexity: O(n^2*L)
    a. n - # of pixels
    b. L - # of pixels in back window

Search Buffer									CP	Lookahead Buffer										Out
									a	b	r	a	c	a	d	a	b	r	a	(0, 0, a)
								a	b	r	a	c	a	d	a	b	r	a		(0, 0, b)
							a	b	r	a	c	a	d	a	c	r	a			(0, 0, r)
						a	b	r	**a**	c	a	d	a	b	r	a				(3, 1, c)
				a	b	r	**a**	c	**a**	d	a	b	r	a						(2, 1, d)
		a	**b**	**r**	**a**	c	a	d	**a**	**b**	**r**	**a**								(7, 4, -)

Conclusion:

Furthermore, while thousands of images are required to correctly train the machine learning model, the time complexity of these algorithms are just too high for the compression ratio it archives.

Even in O(n*L) the compression ratio is just to low for the time taken.

UNIVERSIDAD EAFIT

Obstacles with Huffman Coding

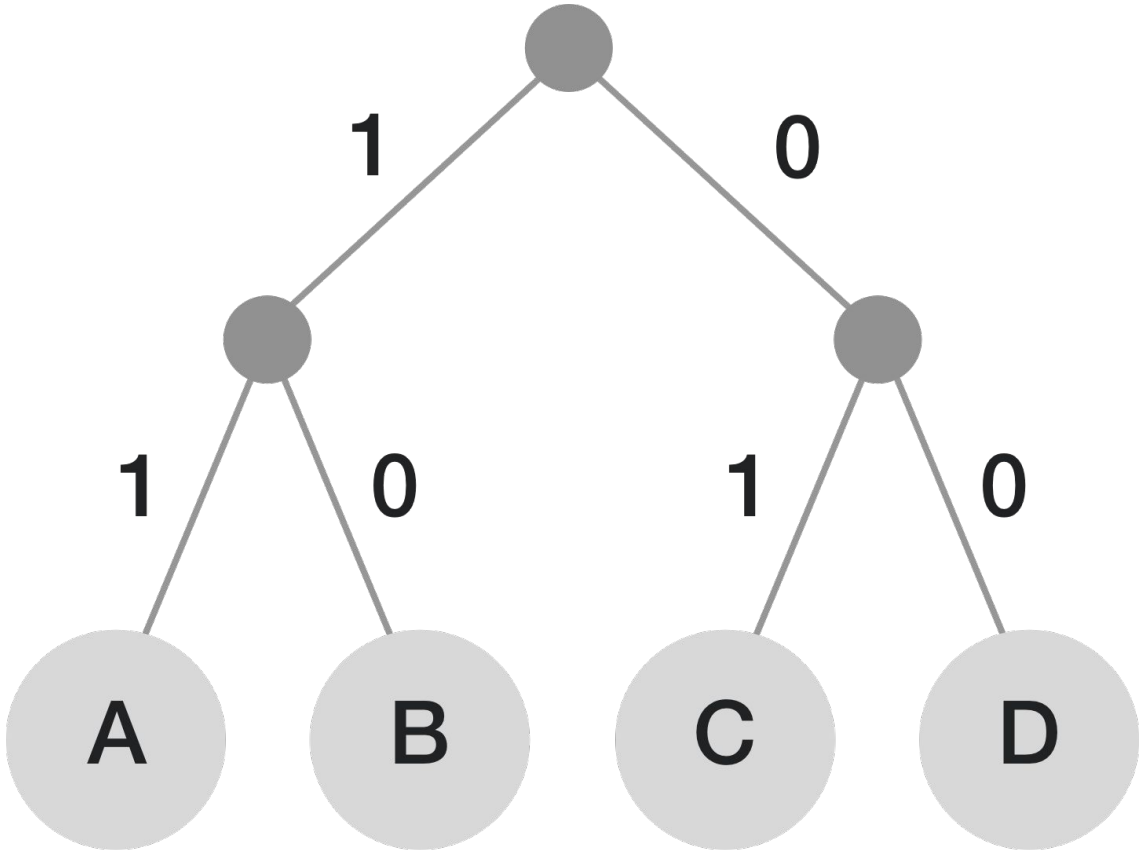
Pixel matrix example:

101,85,90,91,92,93,94,105,105,105

Problems to encounter:

1. Each pixel in the matrix has a low frequency, which means that in average, no pixel repeats itself more than 2% of the time.
2. There are 255 unique values in a pixel matrix.
3. The huffman codes tend to grow larger than 8 bits, which is more than the binary representation of the integers that are supposed to replace.

010101011
010100111111

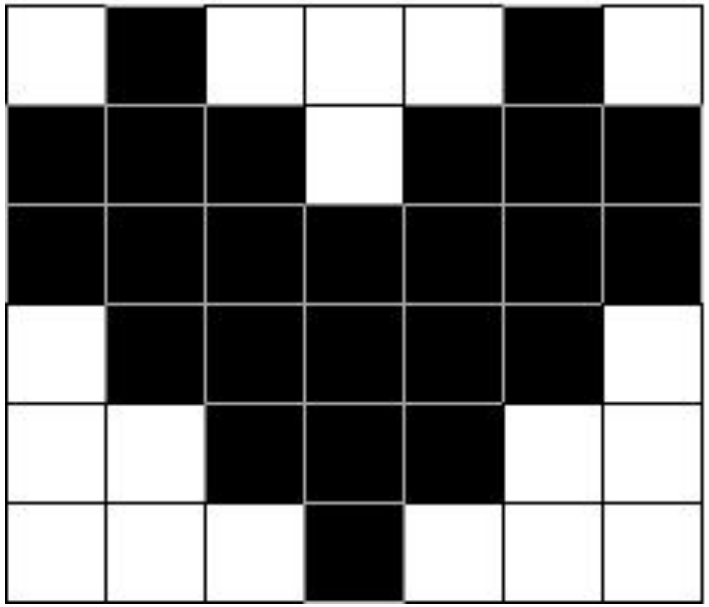


Run-length encoding Algorithm Design

Problems to encounter:

1. This is pretty much, a really straightforward algorithm that works with the frequencies of the same adjacent numbers. The only problem that we could encounter would be if the pixels never repeated.

Example of Image compression using RLE




```
111111111115444444444444333
```

```
12*1,1*5,12*4,3*3
```

LossLess Compression Algorithm

We call this algorithm "Restas"

The idea is to store the difference between the current pixel and the subsequent pixel. This will ensure that the numbers we are dealing with, become smaller which can help the RLE compression.

`10,40,30,20,10,30,10`  `30,-10,-10,-10,20,-20`

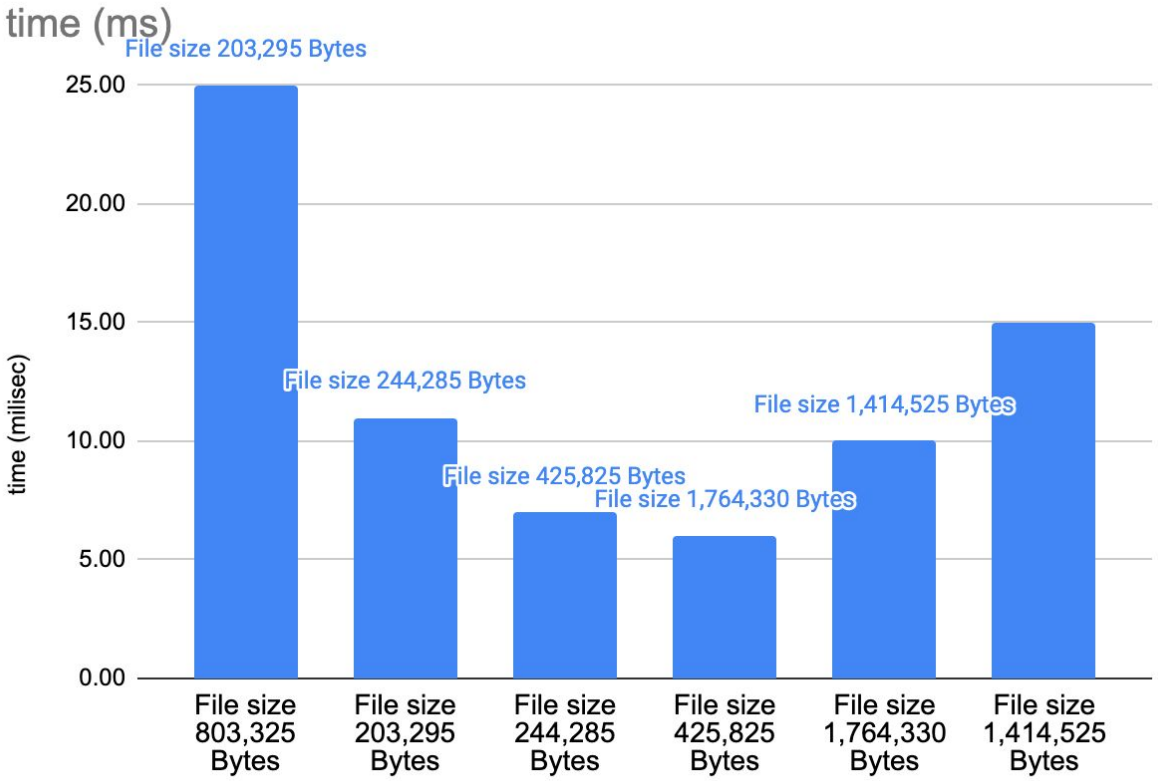
`Here, we can clearly see that this can increase the compression of the RLE.`

Compression Algorithm Complexity

	Time Complexity	Memory Complexity
Image compression	O(N*M)	O(N*M)
Image decompression	O(N*M)	O(N*M)

Run length algorithm combined with summations algorithm. N represents the number of rows in the photograph`s pixel matrix and the M represents the number of columns in the photograph`s pixel matrix.

Time Consumption LossLess



Time Consumption

Main Takeaways:

1) We are able to keep execution times under one second.

2) Low time complexity = low AWS bill.

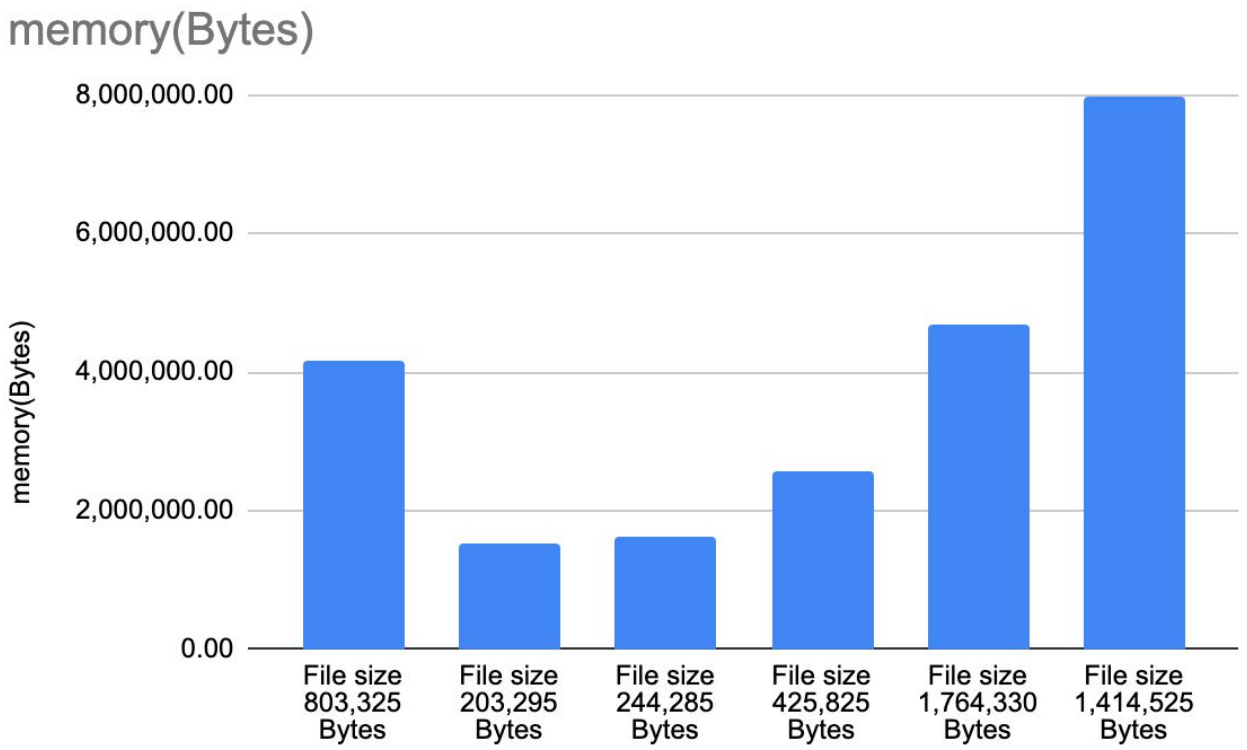
3) The algorithm is able to handle more training data.

UNIVERSIDAD EAFIT

Time Memory Consumption LossLess

Main Takeaways:

1) No more memory that absolutely necessary is needed.
2) No polinomic memory complexity.
3) low AWS bill



Memory Consumption

Average Compression Ratio LossLess

	Compression Ratio
Healthy Cattle	20 : 13
Sick Cattle	20 : 13

Average compression ratio for Healthy Cattle and Sick Cattle.

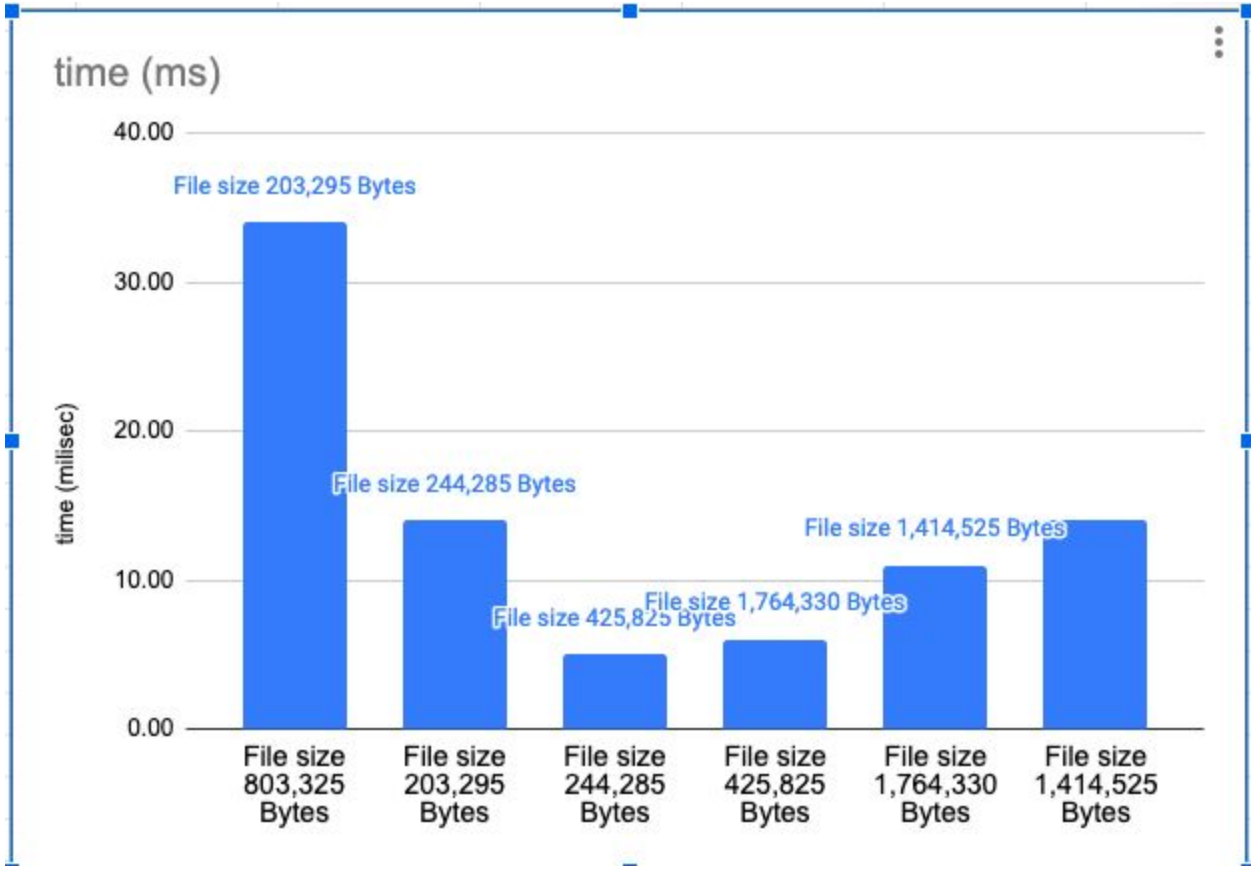
LossLess + Lossy

1. Apply Bilinear Transformation
2. Use the Restas algorithm
3. Run-length Encoding

26,0*23,1,0*3,1,0*3,-2,0*3,1,0*16,-1,0,-1,0,1,0,1,-1,0*3,1,0*3,3,0*11,1,0*11,-1,0*3,1,0,1*2,-1,0*3,-1,0*3,1,-1*2,0,-1,-2,-1,1*2,0*7,-2,1*3,

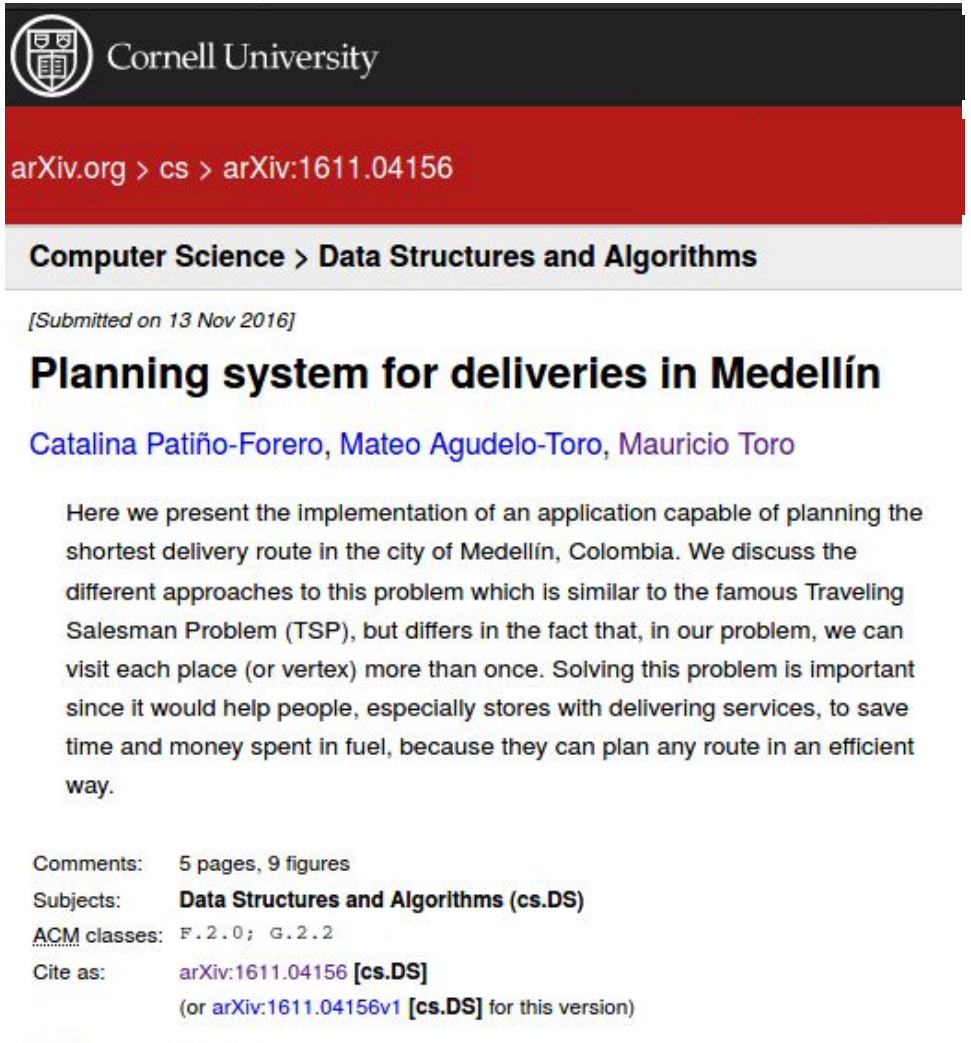
compress ratio: **68 : 11**

Time


time (ms)
40.00
30.00
20.00
10.00
0.00
time (milisec)
File size 203,295 Bytes
File size 244,285 Bytes
File size 425,825 Bytes
File size 1,764,330 Bytes
File size 1,414,525 Bytes
File size 803,325 Bytes
File size 203,295 Bytes
File size 244,285 Bytes
File size 425,825 Bytes
File size 1,764,330 Bytes
File size 1,414,525 Bytes

Report Accepted on arXiv

C. Patiño-Forero, M. Agudelo-Toro, and M. Toro. Planning system for deliveries in Medellín. ArXiv e-prints, Nov. 2016. Available at: https://arxiv.org/abs/1611.04156



Cornell University

arXiv.org > cs > arXiv:1611.04156

Computer Science > Data Structures and Algorithms

[Submitted on 13 Nov 2016]

Planning system for deliveries in Medellín

Catalina Patiño-Forero, Mateo Agudelo-Toro, Mauricio Toro

Here we present the implementation of an application capable of planning the shortest delivery route in the city of Medellín, Colombia. We discuss the different approaches to this problem which is similar to the famous Traveling Salesman Problem (TSP), but differs in the fact that, in our problem, we can visit each place (or vertex) more than once. Solving this problem is important since it would help people, especially stores with delivering services, to save time and money spent in fuel, because they can plan any route in an efficient way.

Comments:	5 pages, 9 figures
Subjects:	**Data Structures and Algorithms (cs.DS)**
ACM classes:	F.2.0; G.2.2
Cite as:	arXiv:1611.04156 **[cs.DS]**
	(or arXiv:1611.04156v1 **[cs.DS]** for this version)

UNIVERSIDAD EAFIT

THANK YOU!

Supported by

The first two authors are supported by a Sapiencia grant financed by Medellín municipality. All the authors would like to thank the "Vicerrectoría de Descubrimiento y Creación", of Universidad EAFIT, for their support on this research