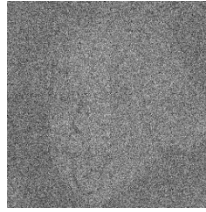


# Guion 6:

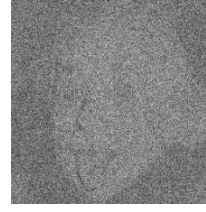
## **Actividad 1:**

Resultados obtenidos (con  $N\_IMGS = 10$ ):

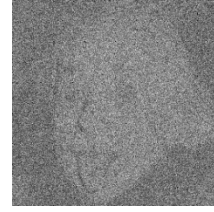
```
TESTING VORAZ:
-ZNCC: 0,033329
-Contador: 10
TESTING BACKTRACKING SIN BALANCEO:
-ZNCC: 0,075752
-Contador: 88573
TESTING BACKTRACKING CON BALANCEO:
-ZNCC: 0,072917
-Contador: 13859
```



*Voraz*



*Backtracking  
sin Poda*



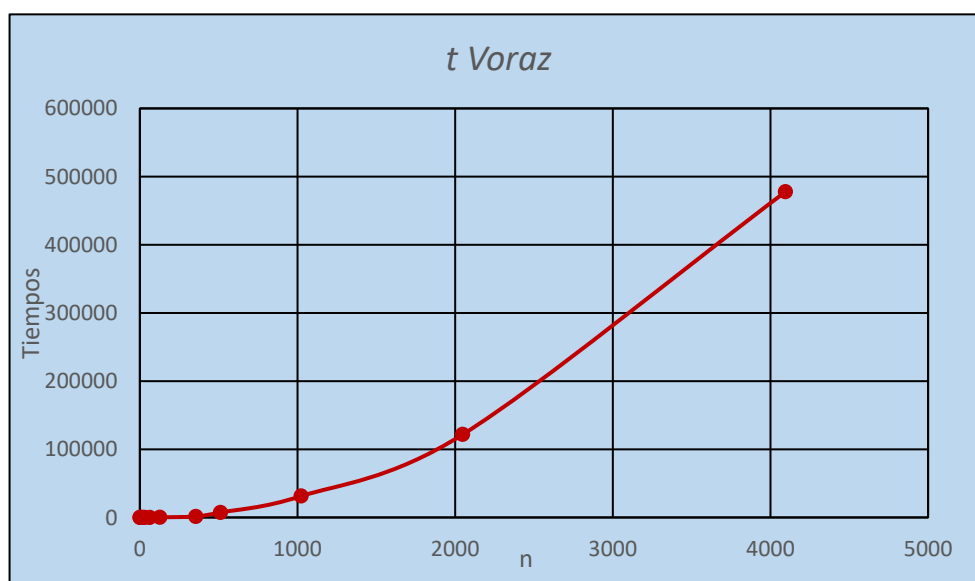
*Backtracking  
con Poda*

## **Actividad 2:**

Medición de Tiempos:

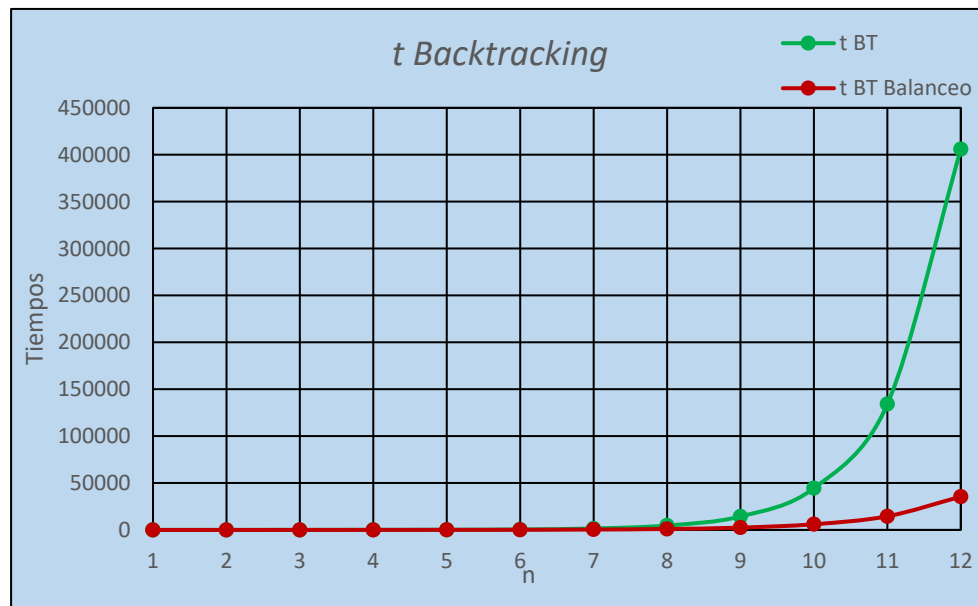
- Voraz (Greedy):

$n$	$t$ Voraz	ZNCC Voraz
2	1	0.0
4	3	0.017467046156525612
8	5	0.03077683597803116
16	15	0.06186828389763832
32	35	0.10464180260896683
64	78	0.17658551037311554
128	280	0.25451239943504333
356	1402	0.38861823081970215
512	7301	0.5400888323783875
1024	31359	0.6777322292327881
2048	121747	0.8030496835708618
4096	477862	0.8862728476524353



- Backtracking:

$n$	$t_{BT}$	$t_{BT \text{ Balanceo}}$	$ZNCC_{BT}$	$ZNCC_{BT \text{ Balanceo}}$
1	2	2	0.0	0.0
2	7	5	2.702029887586832E-4	2.702029887586832E-4
3	22	12	0.026205075904726982	0.019405223429203033
4	72	38	0.03298214450478554	0.03298214450478554
5	187	70	0.0379689559340477	0.0379689559340477
6	504	170	0.04519888013601303	0.04519888013601303
7	1536	407	0.054789576679468155	0.053706031292676926
8	4685	1005	0.05429284647107124	0.051533110439777374
9	14360	2461	0.06933623552322388	0.06925073266029358
10	44498	6043	0.07610578089952469	0.07320989668369293
11	134195	14554	0.09641194343566895	0.09101918339729309
12	405910	35476	0.08351458609104156	0.08132179826498032



## **Actividad 2:**

Respuesta a las preguntas:

- ¿Qué algoritmo proporciona mejores resultados y por qué?  
*Backtracking, porque trata todas las soluciones posibles.*
- ¿Qué algoritmo usarías para procesar un conjunto de datos con un millón de imágenes? Explica por qué.  
*Voraz, debido a que el tiempo que tardaría en devolver la imagen resultado sería mucho inferior con respecto al Backtracking.*

- Determina la complejidad temporal del algoritmo Backtracking sin considerar la condición de balanceo. Valida este análisis utilizando las medidas experimentales.

$O(3^n)$ .

Tomamos  $n_1 = 11$  y  $n_2 = 12$ , con  $t_1 = 134195$ , para demostrar que la complejidad del algoritmo es  $O(3^n)$  aplicamos la siguiente formula:

$$t_2 = \frac{f(n_2)}{f(n_1)} * t_1 = \frac{3^{n_2}}{3^{n_1}} * t_1 = \frac{3^{12}}{3^{11}} * 134195 \Rightarrow t_2 = 402585 \text{ s}$$

Como el tiempo resultante es aproximado al que se ha obtenido en la gráfica queda demostrado que la complejidad del algoritmo es  $O(3^n)$ .

- En términos de tiempo, ¿es ventajoso incluir la condición de balanceo? ¿afecta esta condición a la calidad de los resultados?

*Si, respecto al tiempo es ventajoso, ya que este se reduce, sin embargo, la calidad de los resultados también se ve afectada, teniendo así un ZNCC menor con respecto al Backtracking sin dicha condición.*