Universitat Autònoma de Barcelona

Dipòsit digital de documents de la UAB

This is the **published version** of the bachelor thesis:

Quintas Torra, Pau; Bartrina Rapesta, Joan, dir. Eina visual per la compressió d'imatges. 2023. (Enginyeria Informàtica)

This version is available at https://ddd.uab.cat/record/280684

under the terms of the BY-NC-ND license

Eina visual de compressió

Pau Quintas Torra

Resum— La compressió d'imatges és una presència constant en el nostre dia a dia. Hi ha diversos llocs web que permeten tan la compressió com la reducció del pes de les imatges. Tot i això, cap d'elles t'acompanyen en el procés de la compressió, fet que limita l'aprenentatge de l'usuari. Aquest treball presenta una eina visual de compressió que permet a l'usuari introduir el procés de compressió amb el qual desitja reduir el pes de la seva imatge, i a la vegada, visualitzar les estadístiques i mètriques que es calculen durant aquest procés. Això permet a l'usuari aprendre sobre les accions que s'estan duent a terme i observar-ho de manera visual. La plataforma està programada amb Angular al front-end i Node.js al servidor, facilitant una connexió fluida entre aquestes dues tecnologies, que actualment són àmpliament utilitzades. Gràcies a aquesta pàgina, s'ha estat capaç de comprendre quins procediments afecten visualment a una imatge i quins no, així com les dades que s'obtenen en executar cada un d'ells. Finalment, s'ha arribat a la conclusió que aquesta eina visual de compressió és un pas important per a comprendre què succeeix quan una imatge és comprimida i com és modificada visualment durant aquest procés

Paraules clau—Wavelet, PSNR, Entropia, compressió, RGB, imatge, codificador per entropia, operació aritmètica

Abstract—Image compression is a constant presence in our daily lives. There are various websites which allow the compression of images as well as reduction of their file size, however, none of them guide you through the process of compression, limitating the potential learning experience of the user. This project introduces a visual compression tool that allows users to input the compression process they wish to apply to their images and, at the same time, visualize the statistics and metrics calculated during the process. This enables users to learn about the actions being performed and observe them visually. The platform is programmed using Angular on the client-side and Node.js on the server-side, facilitating smooth communication between these two widely used technologies. This webpage provides insight not only on the procedures which visually affect an imatge but also those who do not. The data obtained from executing each of them is also supplied. Altogether, this visual compression tool is a significant step in understanding the processes involved in compressing an image.

Index Terms—Wavelet, PSNR, Entropy, compresion, RGB, image, entropy encoding, arithmetic operation

1 Introducció - Context del treball

ACTUALMENT, hi ha diferents llocs web on es pot inserir una imatge i comprimir-la en el format desitjat. Alguns, fins i tot, permeten no només decidir quin sistema de compressió es vol utilitzar sinó que també es pot decidir la qualitat final de la imatge. El gran inconvenient que presenten és que no ofereixen la possibilitat de treballar amb la imatge en un baix nivell.

Avui en dia, existeixen aplicacions, com ara Fiji, que, tot i incorporar eines amb aquestes funcions són molt limitades. Fiji permet treballar amb la imatge i modificar-la tant com es vulgui però no realitzar un procés complet de compressió i veure'n els resultats.

En aquest treball es mostrarà com es pot visualitzar un procés de compressió complet i gràcies a les estadístiques i mètriques que es van calculant durant aquest procés, comparar-les amb les dades extretes d'altres processos executats anteriorment.

La motivació d'aquest treball no és una altra que la creació d'una eina visual de compressió actualment no existent que permet comprimir una imatge de manera personalitzada. D'aquesta manera, l'usuari no només serà capaç de tenir la imatge comprimida de manera personalitzada sinó que també aprendrà de la compressió, ja que podrà usar la mateixa imatge tot canviant alguns paràmetres inserits en anteriors execucions i comparar-ne els resultats obtinguts.

Aquest treball està organitzat de manera que primer es mostraran els objectius amb els quals es vol aconseguir durant aquesta recerca. Posteriorment, s'explicarà a l'estat de l'art el context en el qual es troba aquest treball i quines tecnologies similars a la proposada existeixen. S'esmentarà la metodologia emprada i tot seguit s'explicarà quina ha estat la planificació d'aquest treball. En aquesta planificació s'observarà dos diagrames de Gant els quals permetran tenir una visió d'aquesta planificació molt més clara. Continuarà amb el desenvolupament d'aquesta plataforma. Aquesta secció començarà amb una breu presentació del desenvolupament segons la planificació d'aquest treball. Tot seguit serà exposat el desenvolupament dut a terme tant en el back-end com en el front-end. Posteriorment a aquesta secció s'hi troba una manifestació dels problemes amb els quals aquest treball s'ha anat topant durant el seu transcurs. Després es presentarà l'eina visual de compressió tot explicant quin era el plantejament anterior a la plataforma actual, quines funcions permet i que es pot realitzar amb ella. Abans de finalitzar el document hi ha una secció on es mostraran breument com seran els processos que es duran a terme en la plataforma. Finalment. es mostraran els resultats obtinguts i es compararan amb altres

- *E-mail de contacte: pauquintastorra@gmail.com*
- *Menció realitzada: Tecnologies de la Informació*
- *Treball tutoritzat per: Joan Bartrina-Rapesta (dEIC)*
- *Curs 2022/23*

execucions i imatges per tal de tenir un gran ventall de resultats diferents. Per últim, s'exposaran les conclusions i els agraïments.

2 OBJECTIUS

En aquest apartat es plantejaran els objectius del treball i aquest en són els 4 següents:

1. **Creació d'una eina visual per la compressió:**

   Aquest objectiu està centrat en la creació d'una eina visual web de compressió amb la qual l'usuari és capaç de crear un procés de compressió i de descompressió. Aquest procés personalitzat comprimeix la imatge inserida per l'usuari.

2. **Visualització d'imatges resultants:**

   Un altre objectiu que completa aquesta eina és la visualització de la imatge a mesura que aquesta es va comprimint. Finalment, una vegada acabat el procés de compressió l'usuari és capaç de visualitzar la imatge final. D'aquesta manera té una visió més àmplia del resultat final d'executar el procés i dels canvis que ha anat presentant la seva imatge.

3. **Visualització d'estadístiques i mètriques:**

   La visualització de les estadístiques i les mètriques durant el procés de compressió formen part també dels objectius d'aquest treball. Aquestes dades són mostrades a l'usuari una vegada ha finalitzat el procés. També pot ser descarregat un historial en diferents formats que conté les dades calculades fins al moment.

4. **Validació del procés:**

   Aquest objectiu se centra en la validació del procés creat per l'usuari. Abans de dur a terme el procés al servidor, l'usuari ha de validar aquest procés. D'aquesta manera, s'evita que s'introdueixi un procés erroni tot afavorint al seu aprenentatge.

5. **Comparació de resultat i aprenentatge:**

   Gràcies a les dades recollides fins al moment i a futures execucions l'usuari és capaç de comparar-ne els resultats i aprendre sobre la compressió. Aquest és l'últim objectiu plantejat en aquest treball.

3 ESTAT DE L'ART

Avui en dia s'utilitzen un gran nombre de tecnologies per a fer la vida més senzilla. Aquestes tecnologies treballen amb dades i aquestes necessiten ser rebudes i enviades per

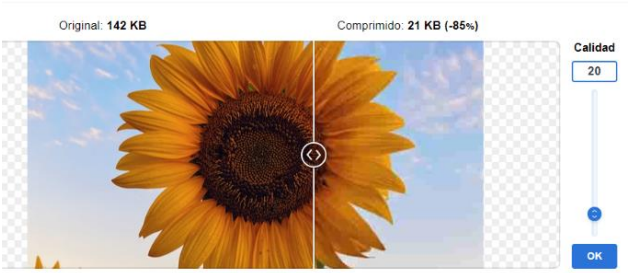
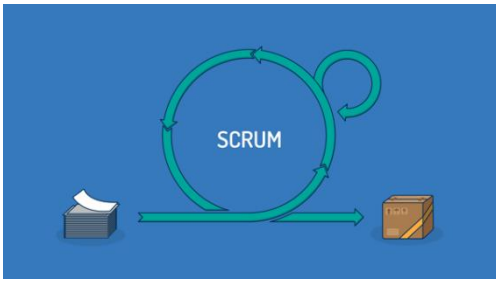


Fig 1 Plataforma web que permet la compressió d'una imatge



Il·lustració 1 Funcionament de la metodologia SCRUM

algú altre. Cada vegada més és necessari que l'emmagatzemat d'aquestes dades ocupi el mínim possible per tal de poder guardar la major quantitat. És necessari que les dades no tinguin un pes elevat i per aquest motiu, la compressió de dades avui dia és molt important.

Si es fa una cerca ràpida, es pot trobar per internet molts llocs web on permeten introduir una imatge i mitjançant uns valors de qualitat redueixen el seu pes. També es troben pàgines que no només permeten introduir una imatge sinó que també t'autoritza veure el resultat de forma visual i comparar-la amb la imatge introduïda. D'aquesta manera l'usuari pot veure si s'aprecia algun canvi visual tot i haver reduït la seva mida (vegeu figura 1). D'altres pàgines simplement permeten introduir una imatge i elles soles s'encarreguen de reduir-li la mida. L'inconvenient que tenen en comú aquestes pàgines és que no es pot decidir res d'aquesta compressió, no és una compressió personalitzada i, per tant, no hi ha la possibilitat que l'usuari pugui aprendre de la compressió ni entendre que és el que s'ha executat per aconseguir aquest resultat. Un restaurant és un exemple semblant a l'esmentat. En un restaurant es rep el plat final i no es té la possibilitat d'aprendre el seu procés, com s'ha cuinat. En aquest treball es desitja que l'usuari sigui capaç d'aprendre sobre la compressió tot fent ús de la plataforma.

Com s'ha pogut observar, no hi ha cap d'aquestes pàgines que permetin escollir els passos intermedis ni introduir una imatge i decidir els processos pels quals aquesta ha de passar de manera detallada, de manera que l'usuari sigui el propietari d'aquesta compressió.

Per altra banda, sí que es troba alguna aplicació que executa a la imatge certs procediments pertanyents a sistemes de compressió actuals com ara la transformada de Wavelet Reversible de Haar. Fiji és una d'aquestes eines, capaç de treballar amb la imatge de manera visual i aplicar aquests

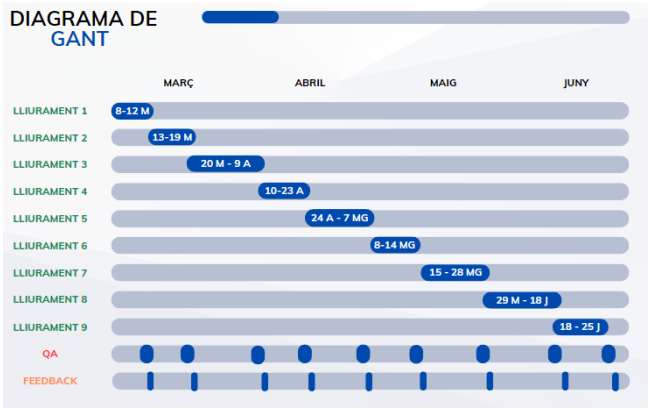


Fig 2 Diagrama de Gant sobre l'antic plantejament

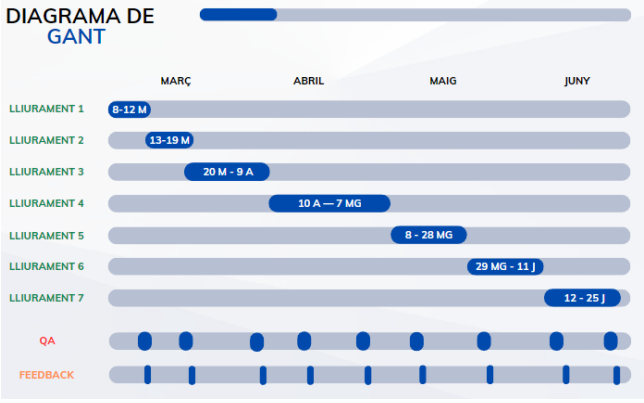


Fig 3 Diagrama de Gant sobre l'actual plantejament del treball

procediments. Però, tot i tenir aquests requisits, no és capaç de comprimir la imatge de manera que es pugui escollir el procés i sigui fàcil de treballar per a l'usuari que no té coneixements sobre la compressió.

En aquest treball es busca crear un procés de compressió fàcil d'entendre i visualitzar, de manera que l'usuari sigui capaç d'obtenir els resultats i comprendre'ls per tal de poder-los comparar amb d'altres.

4 METODOLOGIA

SCRUM és una de les metodologies més utilitzades a l'hora de fer projectes. Permet coordinar, controlar i gestionar l'equip de treball mitjançant un conjunt de valor, principis i pràctiques. Tot i que aquest treball no requereix la gestió d'un equip, l'ús de SCRUM com a sistema d'organització permet mantenir al dia les funcionalitats a implementar. Aquesta metodologia també facilita la producció de diferents versions del codi sense necessitat de finalitzar-lo completament, permetent tenir una versió final funcional, tot i que no sigui la definitiva. A diferència d'altres metodologies que requereixen la programació completa del codi abans de l'entrega al client, SCRUM permet a l'usuari fer proves sobre l'eina visual de compressió sense necessitat de tenir-la completament desenvolupada. Per aquesta raó, s'ha considerat que aquesta metodologia és la més adequada per a la implementació d'aquesta eina web.

Aquest treball s'ha plantejat de manera que l'usuari rep petites versions finals de l'eina. Aquest enfocament permet

Nº	Lliurament
1	Espai de treball
2	Convertidor d'imatges i proves
3	"Let's Create"
4	Compressor d'imatges personalitzat
5	Donar formar (CSS)
6	Explicacions de compressió
7	IA de la imatge
8	Tutorial d'exemple
9	Perfeccionament

Taula 2 Antics lliuraments

Nº	Lliurament
1	Espai de treball
2	Convertidor d'imatges i proves
3	Introducció caixes i creació classes
4	Introducció procés i classes realitzades
5	Validació i procés complet
6	Descarregar historial
7	Refinament i CSS

Taula 1 Nous lliuraments

als usuaris experimentar i beneficiar-se de les últimes actualitzacions i funcionalitats a mesura que es desenvolupa el projecte. Això permet una interacció iterativa i enriquidora entre els usuaris i l'eina, ja que es poden incorporar de forma contínua les millores realitzades. Aquesta planificació ha estat representada en un diagrama de Gant. No obstant això, és important destacar que la planificació inicial no ha estat la mateixa que la planificació final. En la figura 2 s'observa com estaven distribuïts els lliuraments inicials. Aquest canvi va ser degut a l'evolució d'alguns objectius i al temps per a la seva realització. Anteriorment, l'eina visual web de compressió, segons els objectius que estaven plantejats, contenia més d'una secció on cadascuna d'elles realitzava una funció diferent sobre la compressió. En l'apartat on s'explica la plataforma desenvolupada es comenta breument com hagués estat aquesta eina.

En el nou diagrama i taula visualitzables en la figura 3 i taula 1 s'observen els nous lliuraments i les noves dates de finalització de cadascun d'ells. Els dos primers lliuraments coincideixen amb els que s'havien plantejat en la taula 2 tot i que el seu resultat final és diferent. Cal destacar que en aquesta plataforma el convertidor d'imatges no és visible per a l'usuari, ja que s'ha utilitzat per a la realització de proves. Aquestes proves s'han dut a terme fent ús de les noves classes que s'han creat al llarg d'aquest projecte. Gràcies a això, s'ha pogut comprovar que l'execució de la classe i els resultats obtinguts eren els correctes.

El tercer lliurament se centra en la creació de les etapes de la compressió les quals són representades com a caixes, així com el desenvolupament del configurador d'aquestes etapes. A mesura que l'usuari crea aquestes etapes mitjançant el configurador, aquestes es van distribuint fins a formar un procés de compressió. Aquest lliurament també incorpora la creació de certes funcionalitats dels botons, que permeten obrir el configurador d'etapes i afegir imatges. També s'han creat les primeres classes per executar aquestes etapes de compressió en el servidor, són creades en aquest lliurament també. No obstant això, l'usuari no és capaç d'executar un procés de compressió en aquest lliurament a causa de la falta de classes en el servidor i la connexió d'aquestes amb la plataforma.

Aquest quart lliurament és considerat una de les etapes principals i més significatives de tot el treball, ja que en aquesta fase són creades totes les classes necessàries per a dur a terme un procés complet de compressió. En aquest punt, l'usuari és capaç d'introduir un procés complet de compressió i descompressió fent ús dels configuradors de

la plataforma, però en cap cas executar aquest procés complet al back-end. El servidor, tot i tenir totes les classes creades, no està preparat encara per realitzar aquest procés.

Una vegada finalitzat aquest quart lliurament i el següent, s'assolirà l'objectiu principal d'aquest treball, que és la creació d'una eina visual de compressió capaç de comprimir una imatge mitjançant un procés personalitzat.

En el cinquè lliurament hi té lloc la validació o comprovació del procés introduït per l'usuari i l'agrupament al servidor de totes les funcions necessàries per a portar a cap aquest procés. En cas que aquesta validació no sigui l'adequada, l'usuari no podrà executar aquest procés al back-end. D'aquesta manera, l'usuari està obligat a introduir aquest procés correctament.

Com a sisè lliurament es troba la descàrrega de l'historial de les dades recollides de cada execució realitzada per l'usuari. Dins de la pàgina és creat un configurador de descàrrega amb el qual l'usuari obté aquest historial amb el format escollit. Aquest configurador és visualitzable en la figura 4. En aquest lliurament també s'implementen botons per visualitzar imatges i per esborrar les caixes. Aquests botons estan col·locats sota cada caixa inserida per l'usuari. Els botons d'imatge serveixen per mostrar la imatge introduïda amb els canvis visuals que se li han aplicat a l'executar una certa etapa de compressió o descompressió. D'aquesta manera, l'usuari és capaç de visualitzar la imatge després de cada etapa i entendre que és el que se li ha aplicat o modificat. Els botons d'esborrar fan la funció d'eliminar la caixa de compressió o descompressió que tenen a sobre. Gràcies a aquests botons, l'usuari pot canviar l'ordre de les etapes i decidir en quin ordre les vol tenir. També, gràcies a un altre botó, pot veure la imatge recuperada juntament amb les estadístiques i mètriques que s'han anat calculant a mesura que el procés de compressió anava avançant.

Per acabar, el setè i últim lliurament consisteix a revisar el funcionament d'aquesta eina i assegurar-se que sigui l'adequat. També s'implementa CSS a la plataforma per tal que visualment sigui més entenedor.



Fig 4 Configurador de les caixes de compressió a l'eina visual

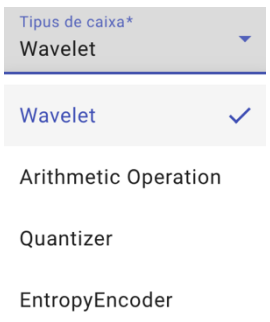


Fig 5 Possibles tipus de caixes de compressió a l'eina visual

5 Desenvolupament

5.1 Programació en el Back-end

En aquest apartat s'explica totes les funcionalitats que s'han anat realitzant en la part del servidor, invisible per a l'usuari però no per això menys important.

Cada possible pas o etapa de compressió té la seva classe al servidor on s'executa el codi. Cadascuna d'aquestes classes conté una funció anomenada "main" la qual s'encarrega d'executar el codi de la classe amb els valors rebuts del front-end.

A part d'aquestes classes també s'han creat unes altres encarregades de dur a terme tota la gestió amb les imatges introduïdes per part de l'usuari. Aquestes classes s'encarreguen d'extreure els valors vermell, verd i blau de les imatges RGB rebudes i les converteix en un objecte de tres arrays de dos dimensions on cadascun d'aquests arrays són els components RGB de la imatge. Aquest objecte és amb el qual es fan les execucions respectives. Posteriorment, es torna a introduir aquestes dades dins d'una imatge per tal que l'usuari obtingui la imatge final comprimida i sigui capaç de veure-ho visualment.

Una de les classes més rellevants d'aquest treball és la classe que controla tota l'execució del procés, anomenada "LetsCreate". Aquesta classe s'encarregarà de gestionar els desitjos de l'usuari (el procés introduït per ell) i usar les classes necessàries per a dur a terme aquest procés amb èxit.

Utils és una classe que serà de molta utilitat durant tot el transcurs del treball, ja que conté funcions que s'aniran fent ús en les diferents classes més d'una vegada.

Per acabar, també es troba una classe encarregada de gestionar l'exportació de les dades emmagatzemades dins d'un fitxer JSON. Aquesta exportació és possible gràcies a dues llibreries que ho permeten realitzar en els formats CSV i Excel.

5.2 Programació en el Front-end

En el front-end s'han implementat diferents components per tal de fer possible el funcionament actual.

El component principal és el que conté tot el funcionament de l'eina visual de compressió. Dins d'aquest component es troben d'altres, com ara el que permet la introducció de caixes de compressió. Aquest s'obre quan l'usuari clica sobre el botó de crear una nova caixa de compressió. El mateix ocorre quan l'usuari clica sobre el botó de crear una caixa de descompressió. El component de crear caixes de descompressió és molt semblant al de compressió, però les dades que s'hi troben dins del component a escollir són diferents. L'últim component que s'ha creat al front-end és l'encarregat de la descàrrega de les dades de les execucions anteriors que té el servidor emmagatzemades. Aquest component permet escollir el format amb el qual l'usuari descarrega les dades i li transmet aquesta informació al component general, que és l'encarregat de dur-ho a terme conjuntament amb el servidor.

6 Plà de contingència

Durant el transcurs d'aquest treball, s'han afrontat diversos problemes que han impedit que la seva realització fos fluida. En aquest petit apartat s'explicarà per sobre quins han estat els problemes més grans que s'han anat trobant



Fig 6 Procés escollit per l'usuari a la plataforma

en aquest treball i com s'han solucionat.

El primer gran problema va ocórrer al principi del treball, a l'hora de crear la primera classe important, la Wavelet. Aquesta classe estava programada amb Python però no de manera correcta per funcionar amb imatges RGB. Es va intentar executar codi Python des del servidor amb Node.js, però com que no se sabia si l'error era del codi programat amb Python o de l'execució d'usar Python en un entorn Node.js, es va optar per una altra solució. Finalment, aquest codi va ser transformat a JavaScript i adaptat per funcionar amb imatges RGB. Una vegada el codi va estar programat amb JavaScript es va aconseguir trobar el problema i solucionar-lo.

Un altre gran problema amb el qual es va topar aquest treball va ser a l'hora de la creació de les caixes tant de compressió com de descompressió. La programació d'aquestes caixes s'ha fet mitjançant el llenguatge d'Angular i es tenia molt poca experiència en aquest llenguatge. A mesura que es va anar avançant en el projecte, la realització d'aquesta es va anar fent més amena.

La creació de l'Excel a l'hora de la descàrrega de les dades en aquest format és un altre gran problema amb el qual aquest treball s'ha topat. La llibreria que permetia aquesta creació no gestionava les dades de manera correcta per tenir els títols desitjats a les taules. Per solucionar aquest problema es va haver de gestionar dada per dada i inserir el títol desitjat.

Per acabar, l'últim gran problema amb el qual aquest treball s'ha topat ha estat a l'hora de realitzar proves amb imatges que tenen dimensió senar. Aquest problema sorgeix quan s'intenta realitzar una transformada de Wavelet, ja que en aquesta etapa de compressió fa una divisió entre dos, de les dimensions de la imatge, i en cas que la imatge sigui de dimensió parella no funciona bé. Es va intentar solucionar forçant que la dimensió senar fos parella restant-li un, però a l'hora de fer proves no s'obtenia el resultat correcte. Finalment, es va solucionar forçant a l'usuari a inserir una imatge de dimensions en potències de dos. En cas que no fos així, la plataforma s'encarrega de transformar la imatge a potència de dos tot inserint informació ja existent a la imatge per tal de solucionar el problema. Aquesta solució no és la manera correcta perquè s'està comprimint més dades de les necessàries, i per aquesta raó s'informa en la plataforma a l'usuari que en cas d'inserir una imatge no adequada (dimensions diferents a potència de dos) els resultats aconseguits no seran del tot certs. En els resultats d'aquest treball es veu algun exemple.

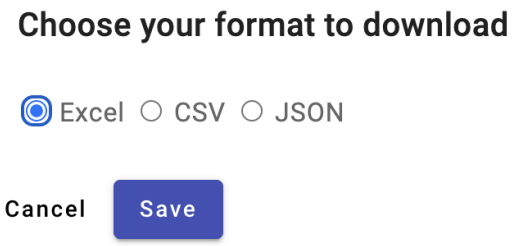


Fig 7 Selector del format per descarregar l'historial de les dades calculades

7 EINA DE COMPRESSIÓ: *LET'S CREATE*

Quan es parla de *Let's create* en aquest treball es refereix a l'eina visual de compressió actual que es planteja en el projecte, tot i que no sempre ha estat aquesta la primera idea. Abans de la seva creació es tenia una idea diferent a la qual s'està plantejant. Primerament. es va pensar en una pàgina web on l'usuari trobés diferents seccions i en cadascuna d'elles experimentés una compressió diferent. Una de les seccions que s'hi hagués pogut trobar és una dedicada a la intel·ligència artificial on l'usuari donaria un parell de valors i gràcies a aquests valors comprimiria al màxim o no la imatge introduïda. Una altra secció estava pensada per tal que fos un convertidor d'imatges, d'aquesta manera l'usuari podria convertir la imatge en el format que desitges. L'última secció que es tenia pensada era la que ara es coneix com a *Let's create*. Aquesta secció era un manipulador d'imatges sense cap mena d'interès en comprimir-la, sinó que l'usuari simplement fos capaç de sumar-li o restar-li un valor a la imatge; aplicar-li diferents tècniques de compressió, com ara la transformada de Wavelet Reversible Haar o fins i tot calcular certes mètriques de la imatge modificada per tal de comparar-les amb la imatge original (tal com ocorre en l'aplicació descarregable Fiji esmentada a l'estat de l'art). Totes aquestes seccions tenien els seus noms respectius. La secció de la intel·ligència artificial s'anomenava AImage, i la secció de manipular la imatge es deia Let's create. Al final aquesta idea va acabar sent modificada i el plantejament del projecte va ser modificat al que és actualment en aquest. En ser un conjunt de la secció del Let's create i afegir algunes idees externes va acabar quedant-se amb aquest nom.

Aquesta eina visual anomenada *Let's create* té moltes funcionalitats i aspectes que veurem a continuació:

En les figures 4 i 5 es visualitza el popup que s'obre cada vegada que l'usuari vol introduir una caixa de compressió. En el cas de voler inserir una caixa de descompressió, el popup que s'obre és molt semblant tot i que dista
dels noms que hi figuren.

Una de les funcionalitats que permet aquesta eina és la introducció d'un procés de compressió i de descompressió seleccionat al 100% per l'usuari. Cada caixa que es mostra dins de la pàgina correspon a un pas de la compressió que s'executarà en el servidor. Els diferents tipus de caixa que pot introduir l'usuari són l'execució de la transformada reversible de Wavelet Haar, el quantitzador, una operació

Wavelet	Quantitzador	Operació Aritmètica	Codificador per entropia	PSNR	Entropia Inicial	Entropia Final	Rati de compressió	Diferencia Entropia (%)
Haar Nivells: 3	Pas de quantització: 3	-	ZIP	40,67	7,35	3,09	2,38	42,04%
Haar Nivells: 5	Pas de quantització: 2	-	ZIP	40,19	7,35	3,64	2,02	49,52%
-	Pas de quantització: 3	-	ZIP	45,95	7,35	5,78	1,27	78,64%
Haar Nivells: 4	Pas de quantització: 4	-	ZIP	35,46	7,35	2,58	2,85	35,10%

Taula 3 Resultats obtinguts a l'executar els processos corresponents amb la imatge del gira-sol

Wavelet	Quantitzador	Operació Aritmètica	Codificador per entropia	PSNR	Entropia Inicial	Entropia Final	Rati de compressió	Diferencia Entropia (%)
Haar Nivells: 3	-	Sumar Valor: 10	ZIP	∞	7,56	4,66	1,62	61,64%
Haar Nivells: 3	-	-	ZIP	∞	7,56	4,66	1,62	61,64%
Haar Nivells: 3	-	Divisió Valor: 3	ZIP	40,49	7,56	2,79	2,71	36,90%
Haar Nivells: 3	-	Multiplicació Valor: 3	ZIP	∞	7,56	4,66	1,62	61,64%
Haar Nivells: 3	Pas de quantització: 3	-	ZIP	40,49	7,56	2,79	2,71	36,90%
Haar Nivells: 5	Pas de quantització: 2	-	ZIP	39,91	7,56	3,32	2,28	43,92%
-	Pas de quantització: 3	-	ZIP	45,91	7,56	5,97	1,27	78,97%
Haar Nivells: 4	Pas de quantització: 4	-	ZIP	35,15	7,56	2,3	3,29	30,42%

Taula 4 Resultats obtinguts a l'executar els processos corresponents amb la imatge del tigre

aritmètica i un codificador per entropia (vegeu figura 5). Aquestes caixes, quan són visualitzades per a l'usuari dins del procés, es mostren cadascuna en un color diferent i amb els valors escollits per ell (vegeu figura 6). La Wavelet té el color verd, el quantitzador té el color vermell, l'operació aritmètica té el color blau i per últim, el codificador per entropia té el color lila. També s'observen amb ordre, ja que l'ordre serà important per a l'execució. Si l'ordre o el procés introduït per l'usuari no és correcte l'execució no es farà efectiva, pel fet que l'eina té una funcionalitat la qual comprova i valida que tot sigui correcte. En cas que no sigui correcte, saltarà una alerta indicant el problema. L'usuari es veurà obligat a corregir-ho, perquè no podrà executar el procés de compressió introduït si no és correcte. Aquesta comprovació la farà fent clic en un botó.

El botó encarregat d'executar tot aquest procés al servidor només serà clicable una vegada l'usuari hagi introduït una imatge a comprimir i l'eina hagi donat el seu vistiplau en la validació. En el moment en què aquest botó sigui clicable i l'usuari dugui a terme aquesta execució al back-end l'usuari restarà a l'espera fins a veure que els botons d'imatge són clicables, que voldrà dir que l'execució haurà acabat.

Una vegada introduït un procés, l'eina et permetrà eliminar una caixa o més per tal que l'usuari pugui canviar

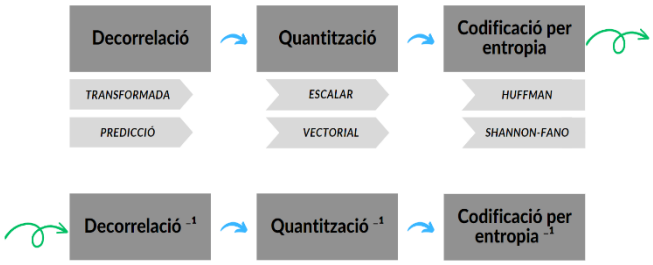


Fig 8 Procés complet de compressió

posicions, valors o altres paràmetres que necessiti. Aquesta altra funcionalitat de la pàgina està sempre activa de manera que abans de comprovar si el procés és correcte podrà eliminar, afegir i refer el procés tant com vulgui. Juntament amb aquesta funcionalitat, sempre estarà present també un botó capaç d'eliminar totes les caixes (tant de compressió com de descompressió) per tal que l'usuari pugui començar de zero i fer una neteja ràpida.

Una altra funcionalitat que es troba en aquesta eina visual web i també és mostrada en la figura 6 és la visualització de la imatge a mesura que el procés va avançant. Una vegada executat el procés, la pàgina desbloqueja els botons d'imatge i permet visualitzar els canvis de la imatge quan passa per una certa etapa introduïda per l'usuari. D'aquesta manera, serà capaç de tenir una idea visual de l'execució i entendrà millor que és el que està executant al servidor.

L'eina, a més a més de tot això, mostra les estadístiques i mètriques necessàries per a fer una bona valoració de la compressió escollida per l'usuari. Aquestes dades són mostrades a la part dreta de la pàgina una vegada el procés ha finalitzat. També permet descarregar-les en diferents formats escollits per l'usuari. Com s'observa en la figura 7 aquests formats són l'Excel, el CSV i el JSON. Aquest popup que es visualitza en aquesta figura es mostra cada vegada que l'usuari decideixi descarregar les dades de l'historial per tal que triï quin és el format amb el qual les vol descarregar. Les dades que descarrega no només contenen les mostrades a la pàgina sinó que són un recull de les últimes execucions realitzades. En cas que no es desitgi descarregar totes les dades, l'usuari haurà d'eliminar les dades que hi són en el fitxer JSON del servidor. Per a fer-ho, l'eina conté un botó específic per a la realització d'aquesta funcionalitat.

8 PROCÉS DE COMPRESSIÓ

Per poder entendre els processos que es duen a terme dins de la plataforma s'ensenyarà com és estructurat un procés de compressió. En la figura 8 s'observen els tres grans passos. La primera etapa del procés és la decorrelació. Aquesta pot ser de tipus transformada, on es troba la transformada Wavelet reversible de Haar, la transformada de 5/3 o la 9/7. En la plataforma es troba la transformada de wavelet reversible de Haar com a exemple d'etapa de decorrelació. La predicció també és un tipus de decorrealció.

Posteriorment, hi ha l'etapa de la quantització la qual pot ser tan vectorial com escalar. En aquest treball, la quantització de tipus escalar és l'escollida per formar part del

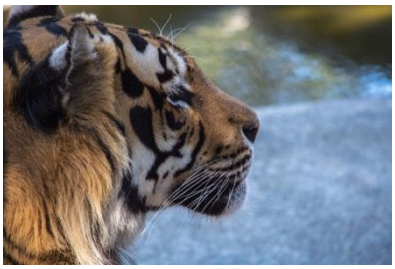


Fig 9 Imatge del tigre



Fig 10 Imatge del gira-sol



Fig 11 Imatge de la Lenna

procés de compressió personalitzat per l'usuari.

Per acabar, la codificació per entropia és l'encarregada de finalitzar aquest procés de compressió. Huffman i Shannon-Fano en són uns exemples d'aquesta etapa. Aquesta eina està dotada amb el ZIP i LZMA com tipus de codificadors per entropia.

Tot i que, actualment a la plataforma, les etapes a escollir per l'usuari són limitades, en un futur es preveu ampliar aquestes per tal que la compressió sigui més estesa.

9 RESULTATS I COMPARATIVA

En aquest apartat del treball es mostraran exemples d'execucions de la pàgina i es compararan els resultats obtinguts. Gràcies als resultats d'aquests exemples s'entendrà millor el funcionament d'aquesta eina i se serà capaç d'aprendre una mica més del món de la compressió.

Per veure els resultats de l'eina visual web s'ha inventat diferents processos de compressió per tal de poder veure la seva execució i les dades aconseguides i ser capaços de comparar-les.

Primer de tot s'ha de crear els processos aleatoris els quals es visualitzen en la taula 4. Aquests processos són inventats de manera no aleatòria, ja que estan preparats per poder ser comentats i extreure'n conclusions sobre els resultats assolits. D'aquests, n'hi ha cinc, més concretament els cinc primers processos, que només s'usaran amb una imatge, perquè ens serviran per explicar com funciona la compressió. Els quatre últims processos són provats amb tres imatges diferents per tal de poder comparar els resultats obtinguts. Les imatges que s'utilitzaran per a realitzar aquestes proves s'observen en les figures 9, 10 i 11.



Fig 12 Imatge modificada per realitzar la transformada Wavelet

Abans de visualitzar i comparar les conclusions caldrà explicar certs valors i conceptes que es veuran en les dades assolides de l'execució dels processos. Aquests conceptes són els següents:

El *Peak Signal To Noise Ratio* (PSNR) és la relació entre la imatge final i la inicial basada en una escala logarítmica. Com més s'assembla la imatge recuperada a l'original més alt és el valor del PSNR. En cas que aquest valor sigui ∞, voldrà dir que s'ha aconseguit reduir la quantitat de dades de la imatge sense perdre la qualitat visual que tenia abans de ser comprimida. Dit d'una altra manera, la imatge final és la mateixa que la imatge comprimida.

També és important entendre el concepte d'entropia, ja que és una dada trobada en els resultats obtinguts. L'entropia és la mitjana de bits necessaris per representar cada píxel de la imatge. L'entropia calculada en aquest treball és l'entropia d'ordre 0. Efectivament, com més baixa sigui l'entropia d'una imatge, millor serà la compressió, pel fet que es necessitaran menys bits per transmetre-la o emmagatzemar-la de manera eficient. Com més redundant sigui una imatge, és a dir, si conté patrons repetitius o informació previsible, més baixa serà la seva entropia i més comprimible serà.

Per finalitzar, és necessari explicar la raó de l'existència de la figura 12. En aquesta figura es presenta una imatge que ha estat modificada perquè tingui dimensions que són potències de dos. Aquesta adaptació és necessària a causa de l'etapa de la transformada de Wavelet reversible de Haar. En aquesta etapa, és convenient que les dimensions de les imatges siguin nombres parells perquè la divisió per la meitat sigui precisa, ja que si les dimensions fossin senars, la divisió no produiria un nombre enter. Tot i que també seria possible aplicar la transformada amb dimensions senars, l'ús de dimensions parells facilita l'execució del procés. En aquest treball, no s'ha emprat la realització correcta d'aquesta transformada. Aquesta eina limita l'ús d'imatges amb dimensions que no siguin potències de dos durant el procés de compressió. Si les dimensions són en potència de dos, la realització de la Wavelet serà la correcta. En cas de trobar-se amb una imatge amb dimensions no adequades, l'eina s'encarregarà de modificar-les per ajustar-les a una potència de dos. Cal tenir en compte que aquesta adaptació resultarà fer una compressió de dades

Wavelet	Quantitzador	Operació Aritmètica	Codificador per entropia	PSNR	Entropia Inicial	Entropia Final	Rati de compressió	Diferencia Entropia (%)
Haar Nivells: 3	Pas de quantització: 3	-	ZIP	40,52	7,28	2,97	2,45	40,80%
Haar Nivells: 5	Pas de quantització: 2	-	ZIP	40,09	7,28	3,59	2,03	49,31%
-	Pas de quantització: 3	-	ZIP	45,92	7,28	5,69	1,28	78,16%
Haar Nivells: 4	Pas de quantització: 4	-	ZIP	34,96	7,28	2,37	3,07	32,55%

Taula 5 Resultats obtinguts a l'executar els processos corresponents amb la imatge de la Lenna

addicionals, ja que s'estarà comprimint una quantitat superior de dades a les quals corresponen originalment.

Totes les execucions usades per explicar el funcionament de la compressió contenen una transforma de Wavelet reversible de Haar de 3 nivells i un codificador per entropia ZIP tal com es mostra en la taula 4. Les diferències entre les execucions es troben a la quantització i a l'operació aritmètica. En els resultats s'observa que aquestes diferències entre els processos no varia el resultat final. Abans d'extreure'n les conclusions després de l'execució dels cinc primers processos, caldrà posar la mirada en les següents dades: el PSNR, l'entropia inicial i final (entropia calculada abans d'enviar les dades codificades), el rati de compressió, i finalment un percentatge que representa la diferència entre l'entropia inicial que s'enviaria i l'entropia final que s'enviarà (un percentatge més baix indica que es requereix transmetre menys dades). En aquesta figura també es consideren els respectius resultats obtinguts després de l'execució de cada procés.

Les primeres conclusions que s'extreuen en executar el procés número 1 i 2 és gràcies al PSNR i l'entropia final, ja que tenen les mateixes dades les dues. Això és degut a la transformada de Wavelet Reversible de Haar, un pas de la compressió que no afegeix pèrdua; tal com indica el seu nom, és reversible. Sempre que es parla d'un procés o pas de compressió reversible significa que no té pèrdua i, per tant, sempre s'aconseguirà que la imatge final sigui exactament igual que l'original. Com s'ha dit anteriorment, s'aconseguirà realitzar un procés on la qualitat visible de la imatge no es veurà perjudicada ni afectada per aquesta compressió. Gràcies a les dades obtingudes es veu que el PSNR és ∞ , per tant, confirma les conclusions esmentades anteriorment.

Tot i tenir una compressió reversible i ,com a resultat, sense pèrdua, no vol dir que no s'hagi comprimit en aquest procés. Com s'observa també en la taula 4, els dos primers processos contenen la mateixa entropia final. Si es mira el procés número quatre també es visualitza aquesta compressió sense pèrdua, ja es torna a assolir un PSNR ∞. A més, conté la mateixa entropia final que els altres dos primers processos, per tant, vol dir que el resultat final de la imatge serà el mateix si s'executa una transformada de Wavelet Reversible Haar sola com si posteriorment s'executa una suma o una multiplicació.

En últim lloc, s'afirma la igualtat a l'hora d'executar els processos número 3 i 5. D'aquesta manera es conclou que la quantització no és res més que una operació aritmètica, en aquest cas, una divisió. També cal destacar la importància del PSNR, ja que no és ∞. Per tant, la imatge final no serà exactament la mateixa que l'original. En aquest procés es veu que la pèrdua aplicada a aquesta imatge és deguda a una divisió dels valors de la imatge entre 3. En aplicar aquest procés s'observa que l'entropia final és menor i el rati de compressió també és més elevat.

Les diverses execucions realitzades per comparar les imatges i avaluar els resultats obtinguts es presenten en les taules 3,4 i 5. A continuació, es detallen les conclusions basades en les execucions realitzades per a la imatge del tigre (consulteu la figura 9), posteriorment per a la imatge del gira-sol (consulteu la figura 10) i finalment per a la imatge de la lenna (consulteu la figura 11).

En la taula 3, on es mostren les dades resultants de l'execució dels processos amb la imatge del tigre, es veu que els dos primers processos a avaluar (procés número 5 i número 6) són bastant similars entre si. No obstant això, el cinquè procés, tot i tenir un PSNR més alt que el segon, presenta una entropia final més baixa. Això implica que en aquesta cinquena compressió no només s'envien menys dades, sinó que la imatge recuperada visualment és més semblant a la imatge original que en el cas de l'execució del sisè procés. D'altra banda, el resultat de realitzar únicament la quantització sense cap etapa prèvia de decorrelació (setè procés a la taula 4) presenta el PSNR més alt de la taula. Tanmateix, mostra una entropia final més baixa i, per tant, un rati de compressió més baix també. Finalment, s'observa el menor PSNR de la taula en el vuitè i últim procés realitzat amb la imatge del tigre. Aquest resultat també indica la major compressió, ja que presenta el rati de compressió més alt. Normalment, quan un procés produeix un PSNR aproximadament inferior a 36 dB, significa que la imatge recuperada mostra canvis visuals que permeten a l'usuari considerar les diferències respecte a l'original. En aquest últim procés, es dona aquesta situació, i per tant la imatge resultant es pot visualitzar a l'apèndix A1.4.

Com es mostra en les taules 3 i 5 les imatges del gira-sol i la lenna presenten resultats similars. Els processos executats en aquestes dues imatges mostren els mateixos patrons que s'observen en la imatge del tigre. La principal diferència entre aquestes tres imatges mostrades radica en la imatge original de la lenna. Aquesta imatge té dimensions que són potència de dos i, per tant, no se li aplica cap mena de correcció. En altres paraules, el procés de compressió no comprimeix més dades de les que seria necessari.

Les imatges recuperades després de l'execució dels últims quatre processos es troben a l'apèndix A1, A2 i A3.

Per a finalitzar aquesta secció, és important analitzar la figura 12. Aquesta imatge representa el resultat de l'adaptació que es realitza al servidor quan s'introdueix una imatge amb dimensions que no són potència de dos. Com s'ha mencionat anteriorment, en el cas que la imatge no compleixi aquest requisit, s'ampliarà fins a la potència de dos més pròxima. Les dades que ompliran les dimensions

afegides seran les mateixes que ja hi figuren a la imatge, però en forma d'efecte mirall, és per això que s'observa el resultat visual que es mostra a la figura 12. En comparar la imatge original (vegeu figura 10) amb la imatge adaptada (vegeu figura 12) s'aprecia aquest efecte mirall del qual s'ha parlat. La imatge original compta amb les dimensions 640 x 853 i a l'hora d'adaptar-la a la potència de dos més pròxima, en aquest cas, 1024 x 1024 s'obté aquest resultat.

10 Ampliació de l'eina

Per tal d'obtenir una comprensió aprofundida dels processos que es duen a terme dins de la plataforma, s'exposarà l'estructura d'un procés de compressió. La figura 8 il·lustra els tres passos principals d'aquest procés. La primera etapa és la decorrelació, que pot prendre la forma d'una transformada com la transformada Wavelet reversible de Haar, la 5/3 o la 9/7. En la plataforma, s'ha seleccionat la transformada Wavelet reversible de Haar com a exemple de l'etapa de decorrelació. La predicció també és un mètode de decorrelació.

A continuació, es troba l'etapa de quantització, que pot ser de tipus vectorial o escalar. En aquest treball, s'ha optat per la quantització de tipus escalar com a part del procés de compressió personalitzat per a l'usuari.

Finalment, la codificació per entropia culmina el procés de compressió. Exemples d'aquesta etapa són els codificadors Huffman i Shannon-Fano. La plataforma està equipada amb els codificadors per entropia ZIP i LZMA.

Cal destacar que, tot i que actualment la plataforma restringeix les opcions de les etapes disponibles per als usuaris, s'està considerant l'ampliació d'aquestes en un futur per aconseguir una compressió més completa.

11 Conclusió

L'eina visual de compressió desenvolupada disposa d'una pàgina funcional on l'usuari pot escollir un procés de compressió i descompressió. Per a definir els processos de compressió/descompressió es poden afegir diferents etapes, com per exemple una transformada wavelet, un quantitzador o un codificador per entropia. Aquest procés introduït s'executa el back-end, desenvolupat en Node.js. La plataforma permet a l'usuari mostrar la imatge en les diferents etapes de compressió i de descompressió per les quals transcorre a més a més de les estadístiques i mètriques calculades durant el procés de compressió. D'aquesta manera l'usuari és capaç de visualitzar l'efecte de la pèrdua introduïda durant el procés de compressió. L'eina desenvolupada també és capaç de mostrar les imatges en les diferents etapes del sistema de compressió definit. Finalment, aquesta plataforma permet a l'usuari descarregar un historial amb les dades calculades durant els processos executats.

Aquest treball ha suposat un desafiament significatiu, ja que s'han fet ús de tecnologies amb les quals hi havia un coneixement limitat. A més, ha consistit en la creació d'una eina inexistent fins ara, la qual cosa ha implicat una manca d'informació disponible.

Aquest estudi de recerca ha estat útil per ajudar a les persones noves en el món de la compressió a entendre de manera visual i interactiva els resultats obtinguts en l'execució de determinats processos de compressió. Tot i que aquesta plataforma no inclou etapes com la predicció o altres transformades de Wavelet, s'encoratja a continuar investigant en aquest àmbit i ampliar aquesta eina per aprofundir en el coneixement d'aquest fascinant món anomenat compressió.

Agraïments

En aquest treball m'agradaria agrair l'ajut en tot moment del tutor, Joan Bartrina, juntament amb el bon tracte rebut i la confiança per a fer aquest treball. També agraeixo a la meva família, parella i amics que m'han acompanyat en aquest procés i donat suport en tot moment.

Bibliografia

[1] "Node.js" node.js.org https://nodejs.org/en (accedit Feb. 2023).
[2] "Express" expressjs.com https://expressjs.com/ (accedit Feb. 2023).
[3] "Lodash" lodash.com https://lodash.com/ (accedit Feb. 2023).
[4] "Angular" angular.io https://angular.io/ (accedit Feb. 2023).
[5] "GitHub" github.com https://github.com/ (accedit Feb. 2023).
[6] "Optimizilla" imagecompressor.com https://imagecompressor.com/es/ (accedit Feb. 2023).
[7] "Compressor online de PNG y JPEG" websiteplanet.com https://www.websiteplanet.com/es/webtools/imagecompressor/ (accedit Feb. 2023).
[8] "Compressornow" compressornow.com https://compressnow.com/es/ (accedit Feb. 2023).
[9] "Angular Material Dialog" angular-university.io https://blog.angular-university.io/angular-material-dialog/ (accedit Març. 2023).
[10] "Angular HTTP Client" angular-university.io https://blog.angular-university.io/angular-http/ (accedit Març.2023).
[11] "Material Dialog" angular.io https://material.angular.io/components/dialog/overview (accedit Març. 2023).
[12] "How to run a Python script from Node.js" mèdium.com https://medium.com/swlh/run-python-script-from-node-js-and-send-data-to-browser-15677fcf199f (accedit Març. 2023).
[13] "W3schools HTML" w3schools.com https://www.w3schools.com/html/default.asp (accedit Març. 2023).
[14] "Extracting a color palette from an imatge with javascript" dev.to https://dev.to/producthackers/creating-a-color-palette-with-javascript-44ip (accedit Abril. 2023).
[15] "Introducción a Express/Node" mozilla.org https://developer.mozilla.org/es/docs/Learn/Server-side/Express_Nodejs/Introduction (accedit Abril. 2023).
[16] "Document: createElement() method" mozilla.org https://developer.mozilla.org/en-US/docs/Web/API/Document/createElement (accedit Abril. 2023).
[17] "Multer" npmjs.com https://www.npmjs.com/package/multer (accedit Abril. 2023).
[18] "Sharp" npmjs.com https://www.npmjs.com/package/sharp (accedit Abril. 2023).
[19] "Jimp" npmjs.com https://www.npmjs.com/package/jimp (accedit Abril. 2023).
[20] "Mathjs" npmjs.com https://www.npmjs.com/package/mathjs (accedit Abril. 2023).
[21] "Lzjs" npmjs.com https://www.npmjs.com/package/lzjs (accedit Maig. 2023).
[22] "Adm-zip" npmjs.com https://www.npmjs.com/package/adm-zip (accedit Maig. 2023).
[23] "How to work with zip files in node.js" digitalocean.com

https://www.digitalocean.com/community/tutorials/how-to-work-with-zip-files-in-node-js (accedit Maig. 2023).
[24] "Json2csv" npmjs.com https://www.npmjs.com/package/json2csv (accedit Juny. 2023).
[25] "Xlsx" npmjs.com https://www.npmjs.com/package/xlsx (accedit Juny. 2023).
[26] "How to convert JSON to CSV in JavaScript" codingbeautydev.com https://codingbeautydev.com/blog/javascript-convert-json-to-csv/ (accedit Juny. 2023).
[27] "Angular Material Radio button" angular.io https://material.angular.io/components/radio/overview (accedit Juny. 2023).
[28] "Transposing a 2D-array in JavaScript" stackoverflow.com https://stackoverflow.com/questions/17428587/transposing-a-2d-array-in-javascript (accedit Juny. 2023).

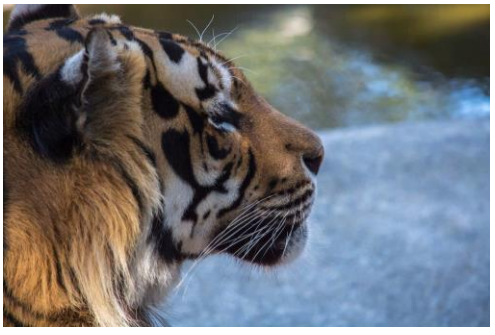
APÈNDIX

A1. Imatges compressió del tigre

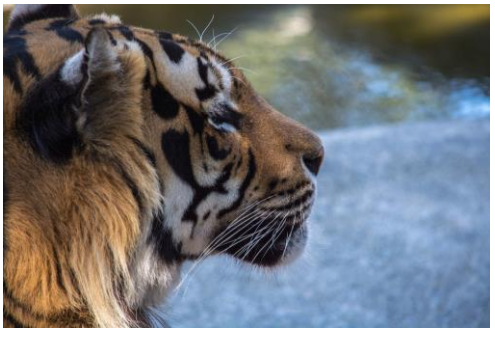
A1.1. Imatge tigre procés número 5



A1.2. Imatge tigre procés número 6



A1.3. Imatge tigre procés número 7



A1.4. Imatge tigre procés número 8



A2. Imatges compressió del gira-sol

A1.1. Imatge gira-sol procés número 5



A1.2. Imatge gira-sol procés número 6



A1.3. Imatge gira-sol procés número 7



A1.4. Imatge gira-sol procés nmero 8



A3. Imatges compressió de la lenna

A3.1. Imatge lenna procés número 5



A3.2. Imatge lenna procés número 6



A3.3. Imatge lenna procés número 7



A3.4. Imatge lenna procés número 8

