

# Informatika Ingeniaritzako Gradua

## Konputazioa

Gradu Amaierako Lana

---

# **Kamera polarimetriko bidezko markatzaile naturalen detekzioa**

---

Egilea

*Iñigo Auzmendi Iriarte*

2022



Informatika Ingeniaritzako Gradua  
Konputazioa

Gradu Amaierako Lana

---

**Kamera polarimetriko bidezko markatzaile  
naturalen detekzioa**

---

Egilea

*Iñigo Auzmendi Iriarte*

Zuzendariak

Imanol Herrera Asteasu eta Gorka Azkune Galparsoro



---

## Laburpena

---

Azken urteotan, ikusmen artifiziala, izandako garapen handiagatik, geroz eta erabiliagoa izaten ari da industria arloan. Lan honetan mundu errealeko industriaren arazo jakin baten aurrean, objektuen detekzioan sakontzen den ikusmen artifizial sistema bat garatu da. Sistema honen helburua, hegazkinen muntaketa garaian bere fuselajeaz ezarri beharreko zuakerrak modu egokian lerrokatzeko, zuaker hauetan dauden markatzaile naturalak detektatzea da. Horrez gain, ohiko kamerak baztertuz, kamera polarimetrikoak deituriko kamera berezien onura eta desabantailak aztertu dira, hauekin lortutako irudiekin datuen gehikuntza (*data augmentation*) nola aplikatu daitekeen ikertuz. Horretarako, lehenengo kamera polarimetrikoaren funtzionamendua eta ezaugarriak ulertu dira. Segidan, kamera berezi horrekin ateratako argazkiak, zeinak detektatu beharreko markatzaile motak dituzten, bildu dira datu multzo bat osatuz. Ondoren, objektuen detekzioarako dauden metodoen azterketa sakona egin eta arazo honetarako interesgarrienak diren metodoak sortutako datu multzoarekin erabili dira. Metodo aukeratuen artean, objektuen detekzioarako sortutako *Faster R-CNN* metodoa dago eta horrez aparte, objektuen detekzioa irudien segmentaziotik burutzen duen metodo bat definitu da *HRNet* eredu erabiliz. Hori egitean, ahalik eta zehaztasun handiena bilatuz, metodoen detekzioetatik abiatuta eta teknika klasikoagoak erabilia, markatzaileen zentroa lortzeko prozedurak definitu eta inplementatu dira. Azkenik, proposatutako sistema *Jetson Nano* mikroordenagailu batean erabili da, integrazio prozesuarekin hasiz. Ondorio bezala, emaitzei erreparaturik, markatzaileen detekzioarako proposatutako prozedura baliozkoa dela ikusi da. Frogatutako bi ereduak emaitza onak eman dituzte, baina zentroen detekzio zehatza burutzeko *HRNet* erabili da itzulitako segmentazio mapen informazio gehigarriagatik. Horrez gain, kamera polarimetrikoaren polarizazio fisikak arazo honetarako markatzaileen detekzioan ez duela emaitzak hobetzen gehiegi lagundu frogatu da. Markatzaileen zentroen detekzio zehatzean berriz, markatzaile guztien zentroa detektatu ezin izan den arren, polarizazioa erabiltzen duten irudiek detekzioan lagundu dutela ikusi da. Azkenik, proiektuarekin aurrera egin eta sakontzeko

---

interesgarriak izan daitezkeen hurrengo pausuak definitu dira.

---

# Gaien aurkibidea

---

<b>Laburpena</b>	<b>i</b>
<b>Gaien aurkibidea</b>	<b>iii</b>
<b>Irudien aurkibidea</b>	<b>vii</b>
<b>Taulen aurkibidea</b>	<b>xi</b>
<b>1 Sarrera</b>	<b>1</b>
1.1 Motibazioa . . . . .	1
1.2 Proiektuaren helburuak . . . . .	3
1.3 Dokumentuaren egitura . . . . .	4
<b>2 Oinarri Teorikoak</b>	<b>7</b>
2.1 Kamera Polarimetrikoa . . . . .	7
2.2 Objektuen detekziorako erabilitako teknikak . . . . .	10
2.2.1 Objektuen detekzioa . . . . .	10
2.2.2 Objektu detekziorako tekniken bilakaera . . . . .	14
<b>3 Proiektuaren garapena</b>	<b>25</b>
3.1 Datu multzoaren eraikuntza . . . . .	25
3.1.1 Irudien lorpena . . . . .	25

iii

---

3.1.2	Anotazioak . . . . .	28
3.1.3	Segmentazio mapak . . . . .	31
3.2	<i>Datu gehikuntza</i> teknikak . . . . .	34
3.2.1	Biraketa . . . . .	34
3.2.2	Iraulketak . . . . .	37
3.2.3	Zarata . . . . .	38
3.3	Detekzioa . . . . .	39
3.3.1	Datu multzoaren banaketa . . . . .	39
3.3.2	Azpiegitura . . . . .	40
3.3.3	Faster R-CNN . . . . .	40
3.3.4	HRNet . . . . .	43
3.4	Markatzaileen zentroyen zehaztapena . . . . .	44
3.4.1	Zuloak . . . . .	46
3.4.2	Gurutzeak . . . . .	48
<b>4</b>	<b>Esperimentuak eta emaitzak</b>	<b>53</b>
4.1	Esperimentazioa . . . . .	53
4.1.1	Detekzio metodoak . . . . .	53
4.1.2	Detekzio zehatza . . . . .	55
4.1.3	Integrazioa . . . . .	55
4.2	Emaitzak . . . . .	57
4.2.1	Detekzio metodoak . . . . .	57
4.2.2	Detekzio zehatzak . . . . .	63
4.2.3	Integrazioa . . . . .	63
<b>5</b>	<b>Ondorioak</b>	<b>65</b>
5.1	Etorkizuneko lana . . . . .	67



---

## Eranskinak

<b>A</b>	<b>Proiektuaren kudeaketa</b>	<b>73</b>
A.1	Proiektuaren deskribapena eta helburuak . . . . .	73
A.2	Proiektuaren plangintza . . . . .	74
A.2.1	LDE diagrama . . . . .	74
A.2.2	Lan-paketeak . . . . .	74
A.2.3	Mugarriak . . . . .	76
A.2.4	<i>Gantt</i> diagrama . . . . .	77
A.3	Lan metodologia . . . . .	78
A.3.1	Lan eremua . . . . .	78
A.3.2	Bilerak . . . . .	78
A.3.3	Planifikatutako ordutegiak . . . . .	78
A.3.4	Ordu estimazioa eta erabiliak . . . . .	78
A.4	Arriskuak eta hartutako erabakiak . . . . .	80
A.4.1	Entrenamendu luzeak . . . . .	80
A.4.2	Proiektu handiegia . . . . .	80
A.4.3	COVID-19 pandemia . . . . .	81
	<b>Bibliografia</b>	<b>83</b>



---

## Irudien aurkibidea

---

1.1	Hegazkinen fuselajea, bertako zuakerren muntaketa prozesua, prozesuan erabilitako zuakerrak eta hauek dituzten markatzaile naturalen adibideak.	2
2.1	Argiaren 3 elementu nagusiak: anplitudea, uhin luzera eta norabidea.	8
2.2	Polarizazio iragazki baten eragina argi ez polarizatuan.	8
2.3	Objektu detekzioaren prozedura orokorra.	11
2.4	Axis Aligned Bounding Box-ak adierazteko bi moduak.	11
2.5	<i>Intersect over Union (IoU)</i> .	12
2.6	Doitasun - Estaldura kurbaren adibidea.	13
2.7	R-CNN metodoaren urratsak.	16
2.8	<i>AlexNet</i> sarearen arkitektura.	16
2.9	<i>Fast R-CNN</i> metodoaren prozesua.	17
2.10	<i>Fast R-CNN</i> metodoak erabiltzen duen VGG-16 CNN sarea.	18
2.11	<i>Region Proposal Network (RPN)</i> -ren funtzionamendua.	19
2.12	<i>Resnet</i> sareko <i>Residual</i> geruza.	20
2.13	<i>Feature Pyramid Networks (FPN)</i> sarea.	21
2.14	<i>HRNet</i> sarea.	22
3.1	<i>FLIR Blackfly S BFS-U3-51S5P-C</i> kamera polarimetrikoa.	25
3.2	Datu multzoko irudiak lortzeko eszenarioa.	26

3.3	Piezen material desberdinak. Ezkerretik eskuinera: aluminioa, RTM zuntza eta CFRP zuntza. . . . .	27
3.4	Zulo eta gurutze motako markatzaileak hurrenez hurren. . . . .	27
3.5	<i>Axis-aligned Bounding Box</i> eta <i>Oriented Bounding Box</i> hurrenez hurren. . . . .	29
3.6	AABB-en errotaziotik lortutako AABB berria. . . . .	29
3.7	Gurutze eta zuloetarako anotatu diren lau puntuen adibideak. . . . .	29
3.8	Gurutze markatzaileen kasuan, anotatutako 4 puntuetatik lortutako AABB-a. . . . .	30
3.9	Anotatutako 4 puntuetatik, OBB-a lortzeko prozesua. . . . .	30
3.10	Zulo markatzaileen anotazioek definitutako OBB-tik, AABB-ak lortzeko, erabilitako elipsearen parametroak. . . . .	31
3.11	Ardatzak definitzen dituzten bi puntuen arteko erdiko puntuak. . . . .	32
3.12	Ardatzen malda bera duten eta zentroko puntuetatik pasatzen diren $Y'$ eta $Y''$ zuzenak. . . . .	32
3.13	Bi zuzen berriek ebakitzen duten puntua. . . . .	33
3.14	Segmentazio mapak sortu ondoren, lausotze prozesuaren eragina. . . . .	33
3.15	Gurutze eta zulo motako markatzaileei dagokien sortutako segmentazio mapak. . . . .	33
3.16	Malus-en legea jarraituz, argiaren intentsitatearen aldaketa bi polarizazio angeluetako iragazkiak erabiliz. . . . .	35
3.17	$I_0 \cos^2(\alpha)$ funtzioan, kameraren 4 iragazkien angeluen intentsitateak. . . . .	35
3.18	Ezkerrean, irudi originalaren AoP irudia. Erdian, proposatutako eta polarizazioa errespetatzen duen metodoarekin eta ohiko metodo arruntarekin $90^\circ$ biratuta lorturiko AoP irudia. Eskuinean, mundu errealean kamera $-90^\circ$ biratuz lortutako AoP irudia. . . . .	36
3.19	Horizontalki <i>flip</i> bat aplikatzearen esanahia eta prozesu horretan iragazki angeluen aldaketa. . . . .	37
3.20	Ezkerrean irudi originalaren AoP irudia. Eskuinean, polarizazioa errespetatuz proposatutako metodoaz bertikalki irauliz eta iraulketa metodo arrunta aplikatuz lortutako AoP irudiak. . . . .	38

---

3.21 Ezkerretik eskuinera: distirarik gabeko irudia, AoP, DoP eta $S_2$ irudiak. Goiko irudiak zaratarik sartu gabe lortu dira, behekoak berriz, irudi originalari zarata sartuta. . . . .	39
3.22 Segmentazio mapetatik BBox-ak lortzeko prozesua. Goran zulo eta behean gurutze motako markatzaileak. Ezkerretik eskuinera, <i>HRNet</i> ereduak itzulitako segmentazio mapak, mapa binarizatuei <i>Close</i> eragiketa morfologikoa aplikatuta lortutako mapak, horietan aurkitutako siluetak berdez ageri diren irudiak eta azkenik, segmentazio mapetatik lortutako BBox-ak. . . . .	43
3.23 Detektatutako markatzaileari dagokion irudi zatia eta irudi berdina linealki eskalatuta 1024 x 1024 tamainara. . . . .	45
3.24 <i>Gaussian Blur</i> iragazkia aplikatu aurretik eta ondoren. . . . .	46
3.25 Irudi binarizatua eta zuloari dagokion pixelen balioek 255 balioa izan dezaten, balioak alderantzikatuta. . . . .	46
3.26 Kalitate oneko eta kalitatea txarreko irudi binarizatuak hurrenez urren. . . . .	47
3.27 Zuloen zentroen detekzioen adibideak. Gorriz detektatutako zuloaren silueta, urdinez doitutako elipsea eta berdez elipsearen eta beraz, markatzailearen zentroa. . . . .	48
3.28 Beheran maskara aplikatuz lortutako irudi binarizatua eta goran berriz, maskarik aplikatu gabe. Bi kasuetan irudi binarizatuak 0 eta 255 balioak alderantzikatu gabe daude oraindik. . . . .	49
3.29 Irudi binarizatuei <i>Zhang-Suen</i> mehetze algoritmoa aplikatuz lortutako irudiak. . . . .	49
3.30 Irudi batetik detektatutako lerroen angeluen histograma. Gorriz lerro gehien dituzten angeluak adierazten dira (0 eta $\frac{\pi}{2}$ ). . . . .	50
3.31 Gurutzearen zentroen detekzioaren prozesua aplikatzean lortutako emaitzen adibideak. Gorriz eta urdinez gurutzeko lerro bakoitzeko erabilitako puntuak eta berdez estimatutako gurutzearen bi lerroak. . . . .	51
4.1 <i>Jetson Nano</i> mikroordenagailua. . . . .	56

4.2	<i>Faster R-CNN</i> metodoarekin egindako zuloen eta gurutzeen detekzioa goran eta <i>HRNet</i> -arekin egindakoak beheran. BBox gorriak <i>ground truth</i> -a irudikatzen dute, berdeak metodoek detektatutako zuloak eta urdinak berriez, metodoek detektatutako gurutzeak. BBox gaineko zifrak BBox detekzioaren konfiantza maila adierazten du. . . . .	61
4.3	<i>HRNet</i> ereduarekin egindako zuloen eta gurutzeen detekzioa eta BBox-ak lortzeko ereduak itzultitako segmentazio mapa. . . . .	62
A.1	Proiektuaren lanaren LDE diagrama. . . . .	74
A.2	Proiektuko <i>Gantt</i> diagrama. . . . .	77

---

## Taulen aurkibidea

---

3.1	<i>Anchor Box</i> tamaina desberdineko multzoekin <i>Faster R-CNN</i> metodoa entrenatuta lortutako emaitzak. . . . .	42
4.1	Datu gehikuntza teknika desberdinekin <i>Faster R-CNN</i> metodoa erabiliz lortutako emaitzak. . . . .	57
4.2	Datu gehikuntza teknika desberdinekin <i>HRNet</i> metodoa erabiliz lortutako emaitzak. . . . .	57
4.3	Sarrera irudi desberdinekin <i>HRNet</i> eta <i>Faster R-CNN</i> metodoak erabiliz lortutako $mAP_{50}$ emaitzak. . . . .	58
4.4	Sarrera irudi desberdinekin <i>HRNet</i> eta <i>Faster R-CNN</i> metodoak erabiliz lortutako doitasun eta estaldura emaitzak. . . . .	59
4.5	Sarrera irudi desberdinekin eta datu gehikuntza moduan, polarizazioaren fisika errespetatzen duten teknikak eta teknika arruntekin <i>HRNet</i> eta <i>Faster R-CNN</i> metodoak entrenatuz lortutako $mAP_{50}$ emaitzak. . . . .	59
4.6	Sarrera irudi desberdinekin eta datu gehikuntza moduan, polarizazioaren fisika errespetatzen duten teknikak eta teknika arruntekin <i>HRNet</i> eta <i>Faster R-CNN</i> metodoak entrenatuz lortutako doitasun eta estaldura emaitzak. . . . .	60
4.7	Sarrera moduan soilik $I_0$ erabilita eta angelu desberdinen balio guztiak erabilita lortutako $mAP_{50}$ emaitzak. . . . .	60
4.8	Sarrera moduan soilik $I_0$ erabilita eta angelu desberdinen balio guztiak erabilita lortutako doitasun eta estaldura emaitzak. . . . .	60
4.9	Metodo desberdinen entrenamendu eta inferentzia denbora eta memoria erabilpena. . . . .	61

---

4.10	Irudi desberdinen ganean, markatzaileen zentroa zehatz detektatzeko prozedurak aplikatzean detektatu ahal izan diren eta baztertu ez diren irudi ratioa. . . . .	63
4.11	Sarrerako irudi desberdin guztiekin lortutako emaitzak batuz, markatzaile zentroaren detekzioaren eta beraz, baztertu ez direnen ratioa. . . . .	63
4.12	<i>HRNet</i> eredua AWS-ko GPU-arekin eta <i>Jetson Nano</i> mikroordenagailuarekin inferituta lortutako denbora eta memoria erabilerak. . . . .	64
A.1	Proiektuaren mugarriak. . . . .	77
A.2	Proiektuko lan-pakete bakoitzeko estimatutako denbora eta benetan behar izan den denbora. . . . .	79



# 1. KAPITULUA

---

## Sarrera

---

### 1.1 Motibazioa

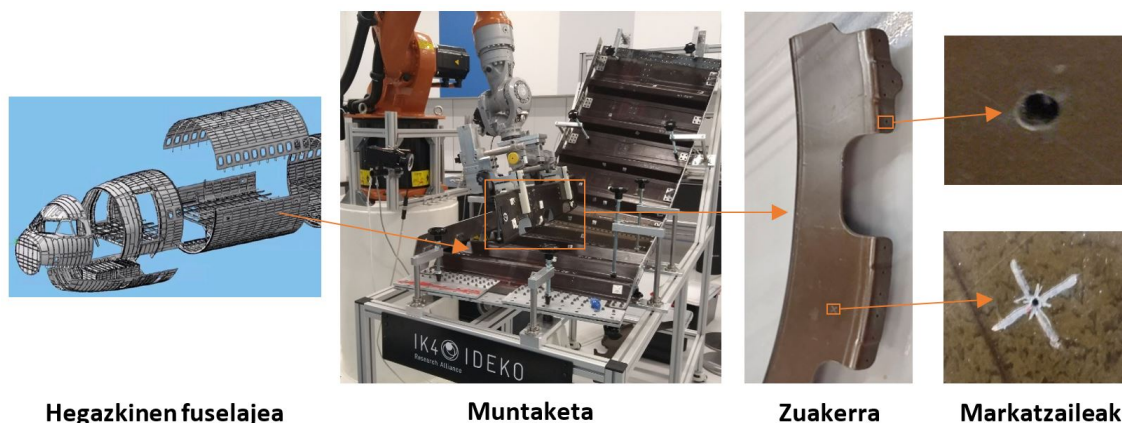
Ikusmena gizakion zentzumen garrantzitsuenetako bat da. Irudiak prozesatzeko gaitasunak gure inguruan zer dagoenaren berri eman eta ingurune horrekin elkar eragitea ahalbidetzen gaitu. Horrez gain, ikusmenak espazioko objektuen posizioaren informazioaren berri ere ematen digu. Adibidez, etxeko atea zabaltzeko, giltza zein posiziotan jarri behar den jakiteko gure ikusmenaz baliatzen gara.

Industrian, oro har, makina robotikoen garapena izugarria izan da, hauen erabilera asko hedatuz. Adibidez, fabrikazio eta muntaketa gune ia guztietan esku robotikoak erabiltzen dira. Hala ere, makina robotiko horiei adimena falta zaie; agindu zehatzez esan behar zaie nora mugitu eta zer egin. Makina hauek ez daukate erabakiak hartzeko ahalmenik eta aurretiko programazio bat behin eta berriz errepikatzen dute. Makina hauei adimena emateko ikusmena gehitzea ez al litzateke aurrera pausu handi bat izango? Ikusmenaren bidez informazio gehiago izango lukete, ohikoak ez diren egoeren aurrean erabakiak hartu eta horren bidezko mugimenduak egiteko.

Azken urteotan neurona-sareek adimen artifizialaren arlo guztietan izan duten garrantzia ezin da ukatu. Konputagailuen potentziaren gorakadak emaitza oso onak ematen dituzten *Deep Learning* eredu interesgarriekin lan egitera ahalbidetu gaitu. Ikusmen artifizialean izandako aurrerakada ere izugarria izan da. Adibidez, kamerak soilik erabiltzen dituzten auto autonomoen lehenengo probak jada abian daude, portaera esperantzagarriak lortuz.

Lan honetan irudien bidezko objektuen detekziorako azken urteotan argitaratu diren eta neurona-sareak erabiltzen dituzten metodoak erabili dira. Metodo hauek askotan erabiliak izan dira emaitzak neurtzeko probetarako datu multzoetan (*benchmark*). Hala ere, mundu errealeko arazoak askotan ez dute zer ikusirik *benchmark* horietako argazkiekin eta beraz, interesgarria da ikustea ea nola funtzionatzen duten metodo hauek industrian erabiltzen diren motako irudi ezohikoekin.

Azken urteetan garatutako ikusmen artifizial prozesuak industrian aplikatzeko helburuarekin, proiektu hau hegazkinen fuselajeen zuakerren muntaketaren automatizazioan laguntzeko ikusmen artifizialeko sistema batean zentratu da. Hegazkinen fuselajeko zuakerren muntaketa burutzen duten makinek zuakerrak zehaztasun handiz parekatu behar izaten dituzte, prozesua modu egokian burutzeko. Horretarako, zuaker bakoitza zehazki non dagoen eta nora mugitu behar den jakin behar da. Hau egiteko, pieza bakoitzaren posizioa identifikatzeko, lehenago eranskailu moduan itsasten ziren markatzaile batzuk erabiltzen ziren. Hala ere, eranskailu horiek zehazki piezen posizio zehatzetan jartzeak lan karga garrantzitsua edukitzen zuen, muntaketa denbora handituz. Eranskailu moduko markatzaileak alde batera utzi eta markatzaile hauek sinpleagoak izatea proposatu zen, gurutze formako eta zuloetan oinarritutako piezen gainazaleko erliebe diren markatzaile naturalak erabiliz. Muntaketa prozesua, bertan erabilitako zuakerra eta hauek dituzten eta detektatu behar diren markatzaileen adibideak 1.1 irudian ikusi daitezke.



Hegazkinen fuselajea

Muntaketa

Zuakerra

Markatzaileak

**1.1 Irudia:** Hegazkinen fuselajea, bertako zuakerren muntaketa prozesua, prozesuan erabilitako zuakerrak eta hauek dituzten markatzaile naturalen adibideak.

Lan honen helburua irudietatik zuakerren posizioa ahalik eta zehatzen lortzea da, baina zehaztasun handia behar den industrian ez da ohikoa neurona-sareak erabiltzea. Sare hauen estokastikotasunagatik eta zurruntasun eta emaitzen azalgarritasun faltagatik ez dira oro har erabiltzen industriaren arlo honetan. Horren ordez, ikusmen artifizialaren tekni-

ka tradizionalak erabiltzea ohikoa da. Lan honetan neurona-sareak arlo honetan sartzeko modu bat definitu da, sareak eta teknika tradizionalak elkartuz.

Proiektu honetan erabili eta aztertu den kamera, ohiko monokromatiko edo RGB izan beharrean, kamera polarimetrikoa da. Normalean, ikusmen artifizialaren arloan RGB kamerak erabiltzen dira eta horien argazkien gainean saiitzen dira ahalik eta emaitza onenak lortzen. Hala ere, arazo batzuentzat badaude beste kamera mota batzuk, polarimetrikoak adibidez, RGB kamerek ez dituzten ezaugarriak dituztenak. Kamera hauek, RGB balioen ordean, polarizazio angelu desberdinen intentsitateak neurtzen dituzte, gainazaletako erliebeen detekzioa hobetuz. Lan honetan kamera berri honen bitartez lortutako irudiak objektu detekziorako erabilitako azken metodoekin nola funtzionatzen duten aztertu da. Honekin batera, ohiko RGB irudiekin alderatu da, erabileran egin beharreko aldaketak kontsideratuz.

## 1.2 Proiektuaren helburuak

Proiektu honen helburu nagusia kamera polarimetrikoaren irudietatik piezetan (zuakerretan) dauden markatzaile naturalen zentroa ahalik eta zehatzen bilatzea da. Helburu nagusia oso orokorra denez, lanaren banaketa gehiago ulertzen laguntzen duten hainbat azpihelburu definitu dira:

- **Kamera polarimetrikoa ulertu:** kamera berezi honek nola funtzionatzen duen jakin behar da. Horretarako, berarekin ateratako argazkien ezaugarriak zein diren ulertu behar da, aurretik literaturan kamera mota hauekin egindako proiektuak begiratzuz eta proiektu honen arazorako zein onura dituen identifikatuz.
- **Datu multzoa sortu:** proiektu honetako objektuak markatzaileak direnez, bere detekziorako ezin daitezke publiko diren ohiko datu multzo handiak erabili. Horrez gain, ohikoak diren datu multzoetako irudiak RGB edo monokromatikoak izaten dira eta beraz, ez dago kamera polarimetrikoarekin ateratako irudirik. Hori kontuan izanda, markatzaile naturalak dauzkaten zuaker multzo bat eta kamera polarimetriko bat erabiliz, markatzaile horiek dituzten argazkiak atera behar dira. Argazkiak ateratzerakoan kameraren parametro desberdinak eta argi-fokuak kontuan hartu beharko dira.

Behin argazkiak aterata, argazki bakoitza anotatzeko programa egokiena eta for-

ma egokiena hautatu beharko dira, ondoren, lortutako irudi guztiak modu erraz eta eraginkorrean anotatzeko.

- **Objektuen detekziorako metodoetan sakondu:** ikusmen artifizialaren teknika tradizionaletatik hasita, gaur egungo neurona-sareak erabiltzen dituzten *Vision Transformer*-etaraino, objektuen detekzioa burutzen duten argitaratutako metodoak zein diren aztertu behar da. Azterketa horretan, bakoitzaren ezaugarriak zein diren, zein prozesu jarraitzen duten, zein motatako sareak erabiltzen dituzten eta nola entrenatu eta erabiltzen diren ulertu beharko da. Ondoren, metodo horietatik, lan honetako objektuen ezaugarriak kontuan hartuta, aipospenak diren metodoak zein diren ikusi beharko da, horietako batzuk hautatuz.
- **Aukeratutako metodoak sortutako datu multzoarekin erabili:** objektuak detektatzeko metodoen artean aukeratutakoak sortutako datu multzoarekin probatu behar dira. Horretarako, aurrez garaturiko metodoak hartu eta entrenamendu eta test prozedurak inplementatzea garrantzitsua izango da, entrenamenduaren eta metodo desberdinen hiperparametroen balio onenak bilatuz. Ondoren, metodo desberdinekin lortutako emaitzekin ondorioak atera behar dira. Azkenean, metodo horietako onena aukeratu beharko da ondorengo prozesuan erabiltzeko.
- **Metodoen emaitzetatik detekzio zehatza burutu:** emaitza onenak eman dituen objektuak detektatzeko metodoaren emaitzekin, markatzaileen zentroa ahalik eta zehatzen lokalizatu behar da. Metodo horien inferentziatik lortutako *Bounding Box*-ek zatitzen duten irudiaren zatia harturik eta ikusmen artifizialeko teknika tradizionalak erabiliz (binarizazioak, filtroak, ingeraden bilaketa...), markatzaileen zentroa detektatu behar da. *Subpixel* mailako detekzio zehatzagoak egiteko, irudi zati hori tamaina handiagora eskalatu beharko da.

### 1.3 Dokumentuaren egitura

Memoria honetan proiektu osoan zehar burututako lanaren informazioa kapitulu desberdinetan banatuta dago. 2. kapituluaren proiektuaren garapenean erabilitako oinarri teoriakoak azaltzen dira, kamera polarimetrikoaren ezaugarri fisikoak, objektuak detektatzeko egon diren metodoak eta emaitzak neurtzeko erabilitako metrikak azalduz. 3. kapituluaren proiektuko garapena zein izan den azaltzen da. Horretarako, datu multzoa nola eraiki den deskribatu eta ondoren, datu multzo horrekin gaur egungo metodoekin nola entrenatu eta

markatzaileak nola detektatu diren azaltzen da. Bukatzeko, azken zentroen detekzio zehatzagoa nola burutu den dokumentatzen da. 4. atalean garapenean definitutako prozedurekin egindako esperimentazioa eta lortutako emaitzak agertzen dira. Azkeneko 5. kapituluan, proiektuaren ondorioak eta proiektuaren etorkizunerako zein ataza burutu beharko liratekeen zehaztu da.



## 2. KAPITULUA

---

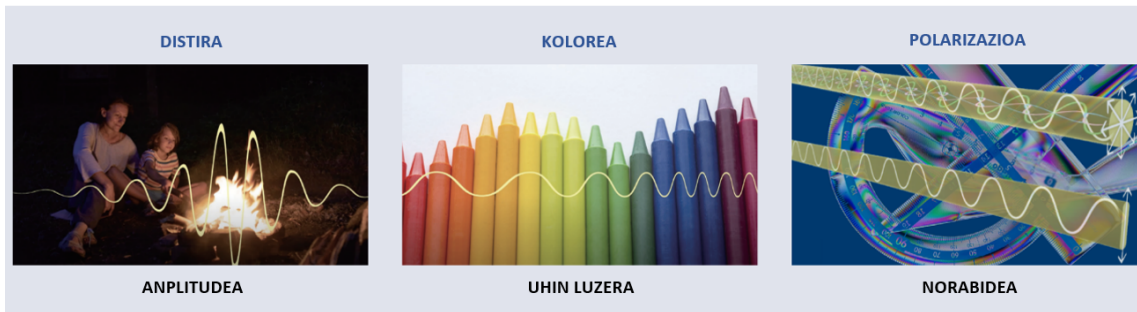
### Oinarri Teorikoak

---

Proiektuaren erronka handi bat kamera polarimetrikoaren erabileran egon da. Kamera berezi hauen ezaugarriak eta funtzionamendua ulertu behar izan dira, eskaintzen dituzten aukerak zein diren eta zertan oinarrituta dauden jakiteko. Behin kameraren funtzionamendua ondo ulertuta, markatzaileak detektatzeko gaur egungo objektuen detekzioerako erabiltzen diren metodo desberdinak aztertu dira. Metodoez gain, detekzioerako ohikoak diren emaitzak neurtzeko metrika desberdinak zein diren ikertu da, bakoitzak informazio desberdina ematen duelarik.

#### 2.1 Kamera Polarimetrikoa

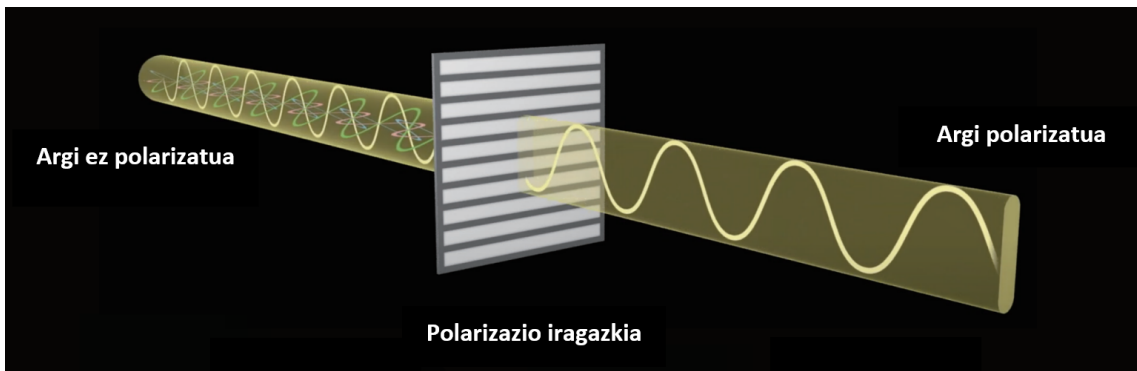
Ikusmena argiarekin oso lotuta dago. Materialek eragiten dituzten argiaren islak ahalbidetzen gaitu gizakioi edozer ikustera; materialek ez balukete argia islatuko, ez genuke ezer ikusiko. Horregatik, garrantzitsua da argiaren ezaugarriak ulertzea. 2.1 irudian ikusi daitekeen bezala, argia uhin elektromagnetiko bat denez, hiru elementu fisiko nagusi ditu: anplitudea argiaren distira kuantifikatzeko, uhin luzera koloreak identifikatzeko eta uhinen dardararen norabidea polarizazioa adierazteko [Goldstein and Collett, 2003].



**2.1 Irudia:** Argiaren 3 elementu nagusiak: anplitudea, uhin luzera eta norabidea.

Eguzkiak edota lanpara fluoreszente batek igortzen duten motako argiaren uhinek norabide askotan dar-dar egiten dutenez, hauek igorritakoari argi ez polarizatua deitzen zaio. Horrela ez bada eta argi uhinek norabide zehatz batean soilik dar-dar egiten badute, argia polarizatua dela esaten da. Distira eta kolorearen kasuan ez bezala, polarizazioa ezin da ohiko kamera batekin edota giza begiekin hauteman.

Kamera digitalentzat polarizazio iragazkiak existitzen dira, hauek kameraren lentsan polarizazio angelu bateko izpiak soilik sartzen uzten dutelarik. 2.2 irudian ikusi daiteke argi ez polarizatuaren izpiak iragazkitik igarotzean polarizatuak bihurtzen direla, bere polarizazio angelua iragazkiaren perpendikularra izanik.



**2.2 Irudia:** Polarizazio iragazki baten eragina argi ez polarizatuan.

Argi ez polarizatua material batean islatzen denean, gainazalaren normalaren eta bere materialaren errefrakzio indizearen menpeko partzialki polarizatutako izpiak islatzen ditu. Islatutako izpien polarizazioa kuantifikatzeko *Stokes* deituriko parametroetan neurtu daiteke:  $S_0, S_1, S_2$ .  $S_0$ -k intentsitate totala deskribatzen du eta  $S_1$  eta  $S_2$  balioek berriz, polarizatutako argia.

*Stokes* parametroetatik bi balio garrantzitsu lortu daitezke: *Angle of Polarization (AoP)* eta



*Degree of Polarization (DoP)*. *AoP*-ak pixel bakoitzeko argia zein orientazioarekin polarizatuta dagoen deskribatzen du eta *DoP*-ak berriz, pixel horretako argi kantitate guztitik zein proportziotan polarizatua den neurtzen du.

*Stokes* parametroen kalkulurako gutxienez 3 polarizazio angelu desberdinen intentsitateak behar dira. Ohiko kamera polarimetrikoek pixel bakoitzeko,  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  eta  $135^\circ$ -etako angelu desberdinetako iragazkiak dauzkate. Kamerak pixel bakoitzeko, polarizazio angelu bakoitzeko, izpiaren intentsitatea neurtzen du. Intentsitate horietatik *Stokes* balioak 2.1, 2.2 eta 2.3 formulekin kalkulatu dira hurrenez hurren [Garcia et al., 2015].

$$S_0 = I(0^\circ) + I(90^\circ) \quad (2.1)$$

$$S_1 = I(0^\circ) - I(90^\circ) \quad (2.2)$$

$$S_2 = I(45^\circ) - I(135^\circ) \quad (2.3)$$

Behin balio horiek kalkulatu, *AoP* eta *DoP*-en balioak 2.4 eta 2.5 formulekin kalkulatu dira [Garcia et al., 2015].

$$AoP = \frac{\arctan\left(\frac{S_2}{S_1}\right)}{2}, \quad AoP \in [0, \pi] \quad (2.4)$$

$$DoP = \frac{\sqrt{S_1^2 + S_2^2}}{S_0}, \quad DoP \in [0, 1] \quad (2.5)$$

Kamera polarimetrikoa erabiliz, orientazio desberdinez osatutako 4 irudi (kameraren iragazki desberdinen angelu haina) lortu daitezke ( $Q_0, Q_{45}, Q_{90}, Q_{135}$  irudiak), hauekin *Stokes* balioak kalkulatu daitezke ( $S_0, S_1, S_2$ ) eta azkenik, *AoP* eta *DoP* balioak eskuratzen dira. Pixel bakoitzeko polarizazio angelu desberdinen intentsitate balioen minimoa hartuz, distirarik gabeko irudia ere lortu daiteke (*GlareRed*). Forma horretan kamera mota hauekin ateratako argazki bakoitzetik, bakoitzak bere ezaugarri desberdinak dituen 10 irudi desberdin lortu daitezke. Ondoren, irudi horietatik lan bakoitzerako egokienak eta erabilgarrienak direnak aukeratu behar dira.

Kamera polarimetrikoak oso erabiliak izan dira medikuntza arloan, emandako irudien kontrastea ohiko kamera monokromatiko edo RGB-en irudiena baino hobea delako [Pierangelo et al., 2013, Qi and Elson, 2017]. Horrez gain, kamera honek dauzkan polarizazio iragazkiek irudietatik nahi ez diren distirak ezabatzen laguntzen dute, objektu gardenen detekzioa ahalbidetuz [Saito et al., 2001, Kalra et al., 2020]. Hala ere, mota honetako

kamerak material desberdinetan trazuak eta akatsak antzematea errazten dutelako aukeratu dira lan honetarako [Lacombe et al., 2019, Lei et al., 2018]. Proiektu honetan detektatu beharreko markatzaileak materialen gainazaleko erliebe bat besterik ez direnez, trazu horietan argiaren islaren polarizazio angelua aldatzen da. Kamera polarimetrikoarekin aldatuta hori neurtu daiteke, trazu edo markatzaileak errazago detektatuz.

## 2.2 Objektuen detekziorako erabilitako teknikak

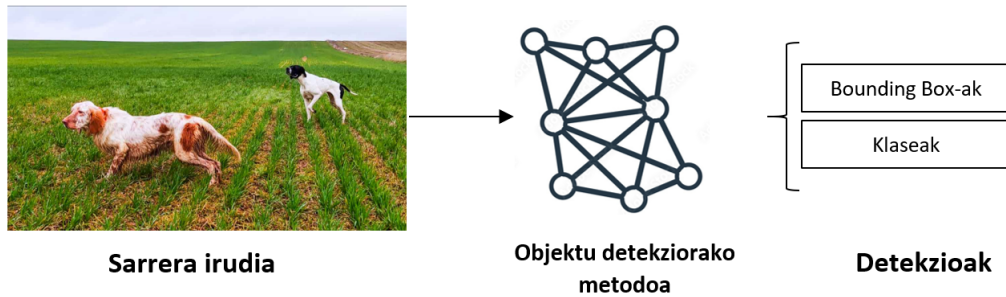
### 2.2.1 Objektuen detekzioa

Objektuen detekzioa ikusmen artifizialeko arazo garrantzitsuenetako bat da. Arazo hau irudi digitaletatik klase zehatz batzuetako objektuak (autoak, pertsonak, animaliak...) hautematean datza, hau da, irudian zein objektu eta non dauden esatea. Objektu detekzioa bi motetan banandu daiteke:

- **Objektu zehatzen detekzioa:** soilik aukeratutako objektuak detektatzen dira. Adibidez, aurpegiaren detekzioa edo autoen detekzioa.
- **Edozein objektuen detekzioa:** giza ikusmena simulatuz, irudi bateko objektu guztiak identifikatzean datza.

Lan honen helburua markatzaile zehatz batzuen detekzioa denez, lehenengo motan zentratu da.

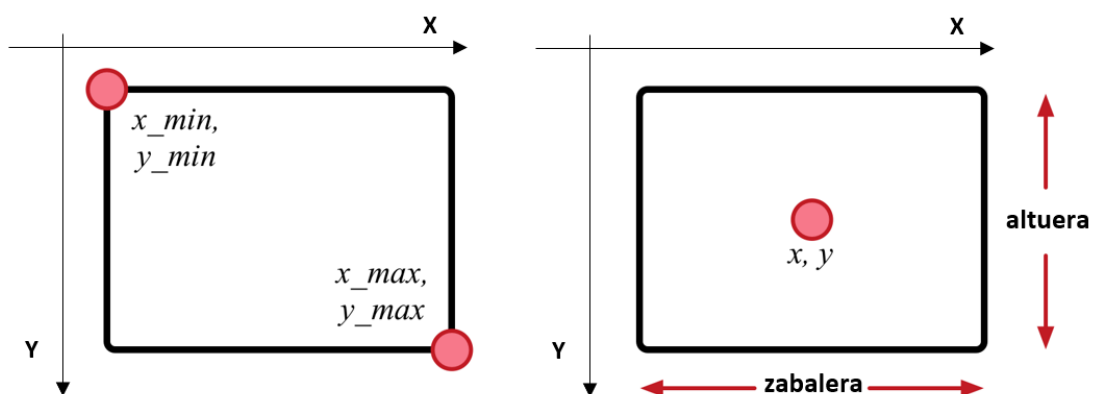
Objektuen detekzioaren prozedura orokorra 2.3 irudian ikusi daiteke. Sarrera moduan irudiak sartzen zaizkio metodo edo ereduari eta irteera moduan bere detekzioak diren *Bounding Box* multzo bat itzultzen du, bakoitzari dagokion objektuaren klasearekin batera. *Bounding Box*-en klaseak definitzeko konfiantza maila bezala ezagutzen den probabilitatea erabiltzen da, hau da, *Bounding Box* hori klase horretakoa izateko probabilitatea.



**2.3 Irudia:** Objektu detekzioaren prozedura orokorra.

### *Bounding Box-ak*

Objektu baten detekzioa adierazteko modua dira *Bounding Box (BBox)*-ak. Hauek sarre-rako irudiaren zati diren laukizuzenak identifikatzen dituzte, detektatu beharreko objektua irudiaren zati horretan dagoela adierazten dutelarik. Ohikoenak direnak eta gaur egungo eredu berriek emaitza moduan ematen dituztenak *Axis Aligned Bounding Box (AABB)* motakoak dira, hau da, irudiaren ardatzekin lerrotatuta dauden laukizuzenak. Hauek definitzeko orokorrean bi punturen koordinatuak erabiltzen dira:  $(x_{min}, y_{min})$  eta  $(x_{max}, y_{max})$ . Puntu hauek 2.4 irudiko ezker BBox-ean ikusi daitekeen antzera, BBox-aren ezker-goiko eta eskuin-beheko puntuak dira hurrenez hurren (beti ere X eta Y ardatzak 2.4 irudikoak izanik).



**2.4 Irudia:** Axis Aligned Bounding Box-ak adierazteko bi moduak.

Hauek definitzeko beste modu bat, BBox-aren zentroaren  $(x, y)$  koordinatuak eta bere zabalera eta altuera gordez da, 2.4 irudiko eskuin BBox-ean ikusi daitekeen bezala.

## Emitzen metrikak

Metodo desberdinen exekuzioetatik lortutako BBox-en kalitatea neurtzeko bi metrika erabili dira. Lehen *mean Average Precision (mAP)*, orokorrean BBox-en zehaztasunean zentratzen dena. Bigarrena detekzioen doitasun eta estaldura, BBox-en zehaztasuna alde batera utziz, *True Positive (TP)* eta *False Negative (FN)*-etan gehiago arreta jartzen dutenak.


- **mean Average Precision (mAP):** objektuen detekziorako metrika erabiliena da eta bere izenak dioen moduan, BBox-en batez besteko zehaztasuna kalkulatzen du. Horretarako, detekzio guztien doitasun eta estaldurak kalkulatzen ditu. Horien kalkulu matematikorako 2.6 eta 2.7 formulak erabiltzen dira.

$$\text{Doitasuna} = \frac{TP}{TP + FP} \quad (2.6)$$

$$\text{Estaldura} = \frac{TP}{TP + FN} \quad (2.7)$$

Non  $TP \equiv \text{True Positive edo Egiazko Positiboak}$ ,  $FP \equiv \text{False Positive edo Positibo Faltsuak}$  eta  $FN \equiv \text{False Negative edo Negatibo Faltsuak}$  diren.

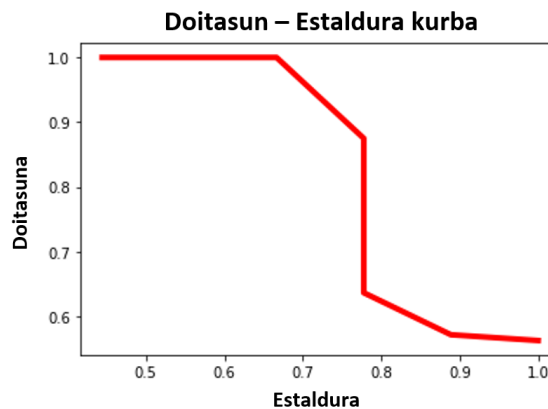
Doitasun eta estalduraren kalkulurako, lortutako detekzioetatik zein diren egiazko positiboak jakin behar da, hau da, zein detekzio asmatu dituen metodoak. Horretarako, mAP-an *Intersection over union (IoU)* erabiltzen da. IoU-ak, 2.5 irudian ikusi daitekeen antzera, ereduak identifikatutako eta errealitatean anotatutako BBox-en artean zenbateko gainezarpena dagoen neurtzen du.

$$\text{IoU} = \frac{\text{Gainezarpn azalera}}{\text{Bateratze azalera}}$$


**2.5 Irudia:** *Intersect over Union (IoU)*.

Detektatutako BBox bat egiazko positibo kontsideratzeko, anotatutako BBox-arekiko IoU-a gutxienez balio zehatz bat baino handiagoa izan behar du. Balio-muga hori 0tik 1erako edozein izan daiteke, oro har 0.5 oso erabilia delarik. Hala ere, askok 0.05 pausoarekin [0.5,0.95] tarteko balio guztiekin mAP-ak kalkulatu eta mAP balio guztien batez bestekoa erabiltzen dute.

mAP-aren kalkulurako, IoU balio zehatz batekin iragazkitako egiazko positiboekin batez besteko zehaztasuna kalkulatu behar da. Horretarako, detekzio guztiak beraien konfiantza mailaren arabera ordenatzen dira, balio handienekoak lehenengoak eta balio baxuenekoak azkenak izanik. Orden horretan, detekzio bakoitzerako uneko doitasuna eta estaldura kalkulatu dira, doitasun-estaldura kurba bat lortuz. Kurba horren adibide bat 2.6 irudian ikusi daiteke.



2.6 Irudia: Doitasun - Estaldura kurbaren adibidea.

*Average Precision (AP)* balioa lortzeko, kurba horren azpiko azalera lortzen da. Prozesu guztia errepikatzen da klase guztietarako, klase guztien AP balioen batez bestekoa mAP balioa izango delarik.

- **Doitasun eta estaldura orokorrak:** mAP metrika oso erabilgarria da orokorrean BBox-en zehaztasuna neurtzeko, baina ez du egoki funtzionatzen detektatu beharreko objektuak oso txikiak direnean. Detektatu beharreko objektuen BBox-en estimazioan pixel gutxietako erroreagatik IoU balioa zeharo jaisten bada, ez baliozko detekzio moduan sailkatzen ditu. Horrez gain, mAP balioak ez du *FP* eta *FN* kopuruaren informazio espezifikorik ematen, bien arteko konbinazioa baizik. Hori kontuan hartuta, 2.6 eta 2.7 formulak erabiliz doitasun eta estaldurak kalkulatu dira. Kasu honetan, ereduak egindako detekzio bat baliozkoa izateko irizpidea, de-

tektatutako eta anotatutako BBox-en zentroen arteko distantzia 5 pixelekoa baino gutxiago izatea erabaki da.

## 2.2.2 Objektu detekziorako tekniken bilakaera

Objektuen detekziorako eta orokorrean irudien prozesaketarako irudietan egindako eragiketa nagusia **konboluzioa** da. Urteetan zehar, konboluzio desberdinen konbinaketak eta beraien iragazkien balioak aztertu izan dira. Hori kontuan hartuta, historikoki bi zatitan banatu ohi dira objektuen detekziorako erabilitako metodoak [Zou et al., 2019]:

- Tradizionala (2014. urtera arte)
- Gaur egungoa (2014. urtetik aurrera)

### Detekzio Tradizionala

Detekzio tradizionalan irudi prozesamendurako teknika klasikoak erabiltzen dira, oro har, adituak daukan problemaren esperientzia baliatuz, eskuz eraturako ezaugarriak erabilita. Konboluzioen kasuan, aurretik eskuz definitutako iragazkiak erabiltzen dituzte, detektatu beharreko objektuetarako onenak diren iragazkiak aukeratuz. Hala ere, detekzio tradizionalak arazoak dauzka oklusioa dagoenean, argiztapen ez uniforme edo itzalak dituzten irudiekin eta beste hainbat egoera konplexuekin. Metodo klasiko ezagunen artean *Viola-Jones Detector* [Viola and Jones, 2001], *Histogram of Oriented Gradients (HOG) Detector* [Dalal and Triggs, 2005] eta *Deformable Part-based Model (DPM)* [Felzenszwalb et al., 2010] lanak daude. Gaur egungo metodoek, metodo tradizional horien hainbat ideietan inspiratutako hainbat prozedura baliatzen dituzte.

### Detekziorako gaur egungo metodoak

Urteak pasa ahala GPUen garapen handiarekin, konputazio ahalmen izugarri handi batera iritsi da. Horrez gain, gordetzen diren irudi eta beste motako datu kantitatea ere erabat hazi izan da azken urteotan. Konputazio ahalmen handi hori erabiltzen duten objektuen detekziorako *Machine Learning (ML)* eta ondoren, *Deep Learning (DL)* ereduak sortzen ari dira etengabe. Eredu hauek normalean datuen anotazioak baliatuz entrenatzen dira (*supervised-training*) eta horregatik, milaka irudi dauzkaten datu multzo asko publikatu dira ereduaren emaitzen neurketak egiteko (*benchmark*-ak). Gaur egungo eredu hauek

egoera konplexuetan oso ongi funtzionatzen dute, detekziorako teknika tradizional guttien emaitzak hobetuz.

Azken urteotan objektuen detekziorako sare erabilienak *Convolution Neural Networks (CNN)*-ak dira. Hauek iragazki desberdinekin konboluzio eragiketak burutzen dituzten hainbat geruza dauzkate, entrenamendua iragazki horien balioak fintzean zentratzen delarik. Detekzio tradizionalan ez bezala, eredu hauetan iragazkien balioak optimizatzen saiatzen dira problema bakoitzean desberdinak izan daitezkeen detektatu beharreko objektuetarako. Entrenamendurako, aurretik anotatutako eta helburuko objektuen irudiez osatutako datu multzoak baliatzen dituzte.

Objektuen detekziorako gaur egungo metodoak bi multzo handietan banatzen dira: *two-stage* eta *one-stage*. *Two-stage* motako metodoek bi urrats ematen dituzte: lehenengo irudia intereseko eskualdeetan banatzen dute eta ondoren, interes eskualde bakoitza sailkatzeko dute. *One-stage* motako metodoek berriz, dena urrats berdinean egiten dute, hau da, BBox-en detekzioa, aurretik eskualde horien banaketarik egin gabe burutzen dute.

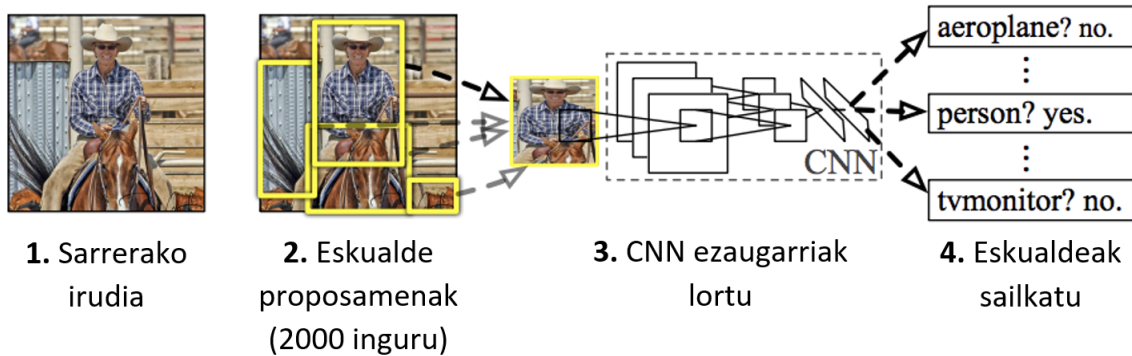
Normalean *two-stage* motako metodoek emandako emaitzak hobekiak dira, BBox zehatzagoak ematen dituztelako. Hala ere, mota honetako metodoek eragiketa asko egin behar izaten dituzte eta beraz, beraien inferentzia denbora altuagoa da. *One-stage* motako metodoek zehaztasun hain ona ez eduki arren eta objektu txikiak detektatzeko arazoak izan arren, bere inferentzia denbora nahiko baxua da, denbora errealeko objektuen detekzioa egiteko aukera emanez. Lan honetan, abiadura baino gehiago zehaztasuna bilatu denez, *two-stage* motako metodoetan zentratu da.

*Two-stage* motako metodoen artean eboluzio argi bat egon da:

- **Region-based CNN (R-CNN)** [[Girshick et al., 2013](#)]: CNN sareak objektuen detekziorako erabiltzen zituen lehenengo metodo funtzionala da.

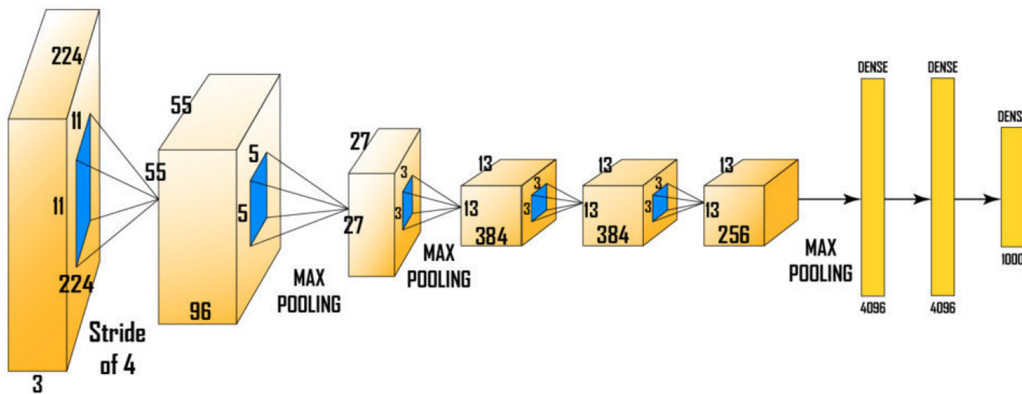
Metodo honek dituen fase desberdinak 2.7 irudian ikus daitezke. Lehenengo, *Selective search* deituriko algoritmo batekin intereseko eskualde desberdinak (2000 inguru) sortzen ditu [[Uijlings et al., 2013](#)]. Ondoren, eskualde bakoitza bereizmen jakin batera eskalatu eta CNN sare batera sartzen da. Sare horrek aukeratutako eskualde bakoitzaren ezaugarri garrantzitsuenak dituen bektore bat itzultzen du. Hori egindakoan, ezaugarrien bektore hori *Support-vector Machine (SVM)* entrenatu batera sartzen da, eskualde bakoitzean, detektatu beharreko klase bakoitza egoteko probabilitatea lortuz. Azkenik, emaitza *Bounding Box Regressor* batera sartzen da, eskualdeetatik BBox-en koordinatu zehatzagoak lortzeko.

## R-CNN: Eskualdeak CNN ezaugarriekin



**2.7 Irudia:** R-CNN metodoaren urratsak.

Bertako CNN-a, hau da, eskualde bakoitzetik ezaugarriak lortzeko erabilitako konboluzioz osatutako neurona-sarea, *AlexNet* [Krizhevsky et al., 2012] sare aurre-entrenatua da. 2.8 irudian dagoen bere arkitekturaren ikusi daitekeen antzera, irudiak bere bereizmena jaisten joaten dira, bere kanal maila eta horrekin, gune bakoitzaren semantika handituz.



**2.8 Irudia:** *AlexNet* sarearen arkitektura.

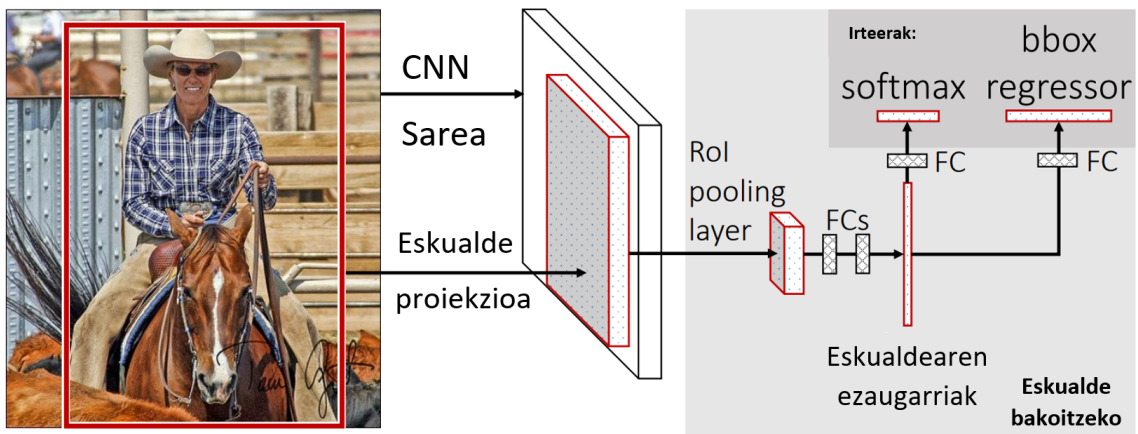
Emaitza onak ematen dituen arren, irudietan aurkitutako intereseko eskualdeak hiru prozesuetatik (CNN, SVM eta BBox regressor) sekuentzialki pasa behar direnez, metodo honen abiadura oso baxua da. Irudi bakoitzetik detekzioak lortzeko 47 segundo behar izaten ditu.

- **Fast R-CNN** [Girshick, 2015]: aurreko *R-CNN* ereduaren aurrera pausoa da. Aurreko erudia oso motela zen, sarrerako iruditik lortutako eskualde bakoitzeko fase



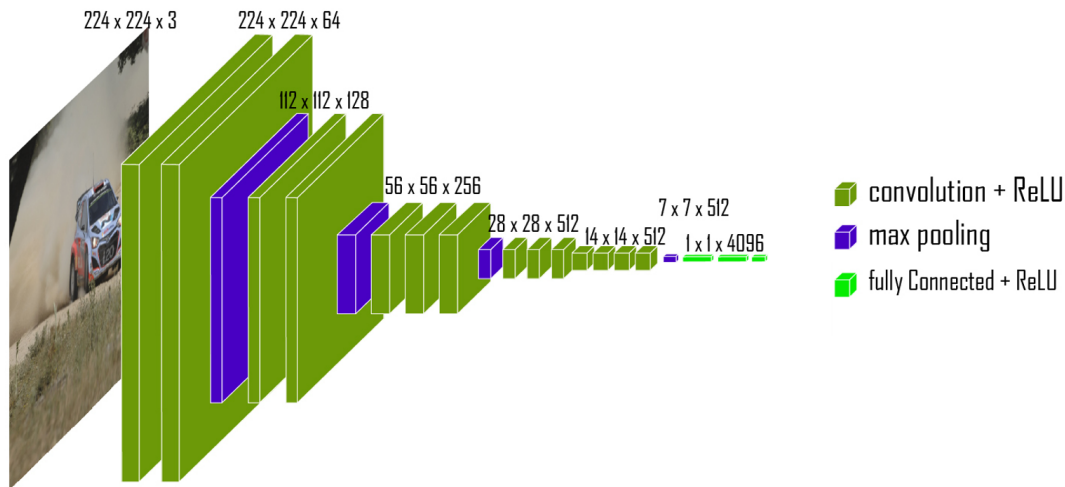
bakoitza sekuentzialki exekutatu eta entrenatu behar zituelako. *Fast R-CNN* metodo berri honek arazo horiek konponduz, abiadura asko handitzen du.

Metodo berri honen prozesua, 2.9 irudian ikusi daitekeen bezala, aurreko *R-CNN* metodoaren oso antzekoa da. Abiadura hobetzeko, iruditik identifikatutako eskualde bakoitza CNN sarera sartu beharrean, sarrerako irudi osoa sartzen da CNN sarera eta honek ezaugarri mapak itzultzen ditu. Eskualdeen aukeraketa, *R-CNN* metodoan bezala egiten da irudi original osoaren gainean, baina ondoren, eskualde horiek irudi osotik lortutako ezaugarri mapa horien dimentsioetara pasa behar dira. Sareak irudien dimentsioak txikitzen dituenaz, lortutako eskualdeen koordinatuekin ere gauza berdina egingo da, bere ezaugarri mapako proiektzioak lortuz. Eskualdeei dagokien ezaugarriak *RoI* motako geruzarekin tamaina finko batera eskalatzen dira. Ondoren, eskualdeen ezaugarri horiekin, detektatu beharreko klase bakoitzaren probabilitateak lortzeko *Fully Connected* motako sare bat erabiltzen da (ez *R-CNN* metodoan egiten zen moduan *SVM* sailkatzaileak erabiliz). Azkenik, *BBox regressor*-a erabiltzen da eskualde horietatik BBox zehatzagoak lortzeko.



**2.9 Irudia:** *Fast R-CNN* metodoaren prozesua.

Aldaketa horiez gain, metodo honetan *R-CNN* metodoan erabilitako *AlexNet* sarea erabili beharrean, geruza gehiago dituen *VGG16* CNN sarea (2.10 irudia) erabiltzen da CNN moduan.

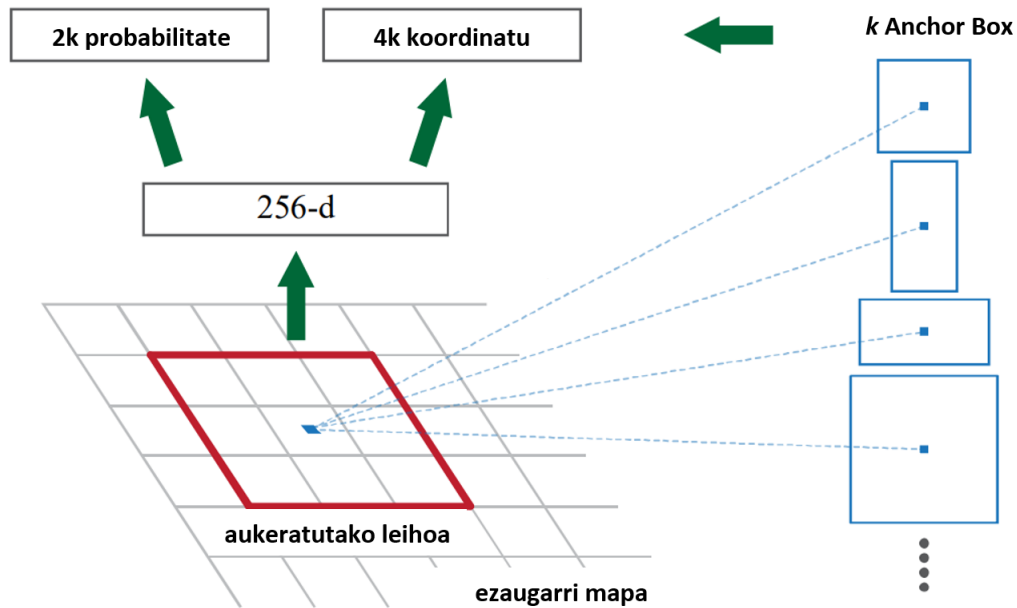


**2.10 Irudia:** Fast R-CNN metodoak erabiltzen duen VGG-16 CNN sarea.

*Fast R-CNN* metodo berri honekin emaitzak hobetzeaz gain, entrenamendu denbora 10 aldiz azkarragoa da eta inferentzia berriz, 146 aldiz azkarragoa.

- **Faster R-CNN [Ren et al., 2015]:** *Fast R-CNN*-aren hurrengo pausoa da. Aurretiko metodoetan, eskualdeen aukeraketa *Selective Search* deituriko algoritmoa irudi originalaren gainean aplikatuz egiten zen. Algoritmo hau CPUan exekutatzen zenez eta CNN sarearekiko independentea zenez, asko moteltzen zuen bai entrenamendu prozesua eta baita inferentzia ere. Horregatik, metodo berri honetan *Region Proposal Network (RPN)* sarea sortu da. Sare honek, irudi osoaren ezaugarri mapak sarrera bezala izanda, eskualdeen proposamenak egiten ditu.

*RPN* sarea *Fully Connected* motako sare txiki bat da. Bere funtzionamendua 2.11 irudian ikusi daiteke. Sarearen konfigurazioan *Anchor Box* bezala ezagunak diren BBox-en tamaina eta aspektu ratio multzo posibleak definitu behar dira. Sareak sarrera moduan ezaugarri maparen leiho finko bat hartzen du eta leiho horretan, definitutako *Anchor Box* bakoitzean, objekturik egoteko edo ez egoteko probabilitateak (ez du zehazten zein objektu, soilik edozein objektu dagoen ala ez) eta objektu horien BBox-aren koordinatuak itzultzen ditu.  $k$  *Anchor Box* definitzen badira, ezaugarri maparen leiho bakoitzeko  $2k$  balio itzuliko ditu *Anchor Box* bakoitzean objekturen bat egoteko eta ez egoteko probabilitatearekin eta  $4k$  balio *Anchor Box* bakoitzeko proposatutako BBox-en koordinatuekin. *Faster R-CNN* metodoko CNN sare nagusitik lortutako irudi osoaren ezaugarri mapan zehar pasatuko da *RPN* sarea, objektuak dauden eskualdeen koordinatuak gordez.



2.11 Irudia: Region Proposal Network (RPN)-ren funtzionamendua.

Behin RPN sarearekin eskualde proposamenak lortuta, *Fast R-CNN* metodoan egiten zen antzera, ezaugarri mapetako eskualde horien balioak pasatuko zakizkie *BBox Regresor* eta klaseen identifikazioa burutzen duten zatiei.

*Faster R-CNN* metodoak aurreko metodoak baino emaitza hobeak lortzeaz gain, 10 aldiz hobetzen du bere inferentzia denbora.

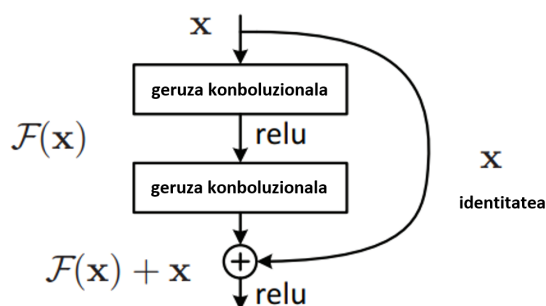
*Two-stage* motako metodoen bilakaera handia ikusirik, lan honetan *Faster R-CNN* azken metodoa erabili da, beste guztien artetik emaitza onenak eman eta entrenamendu eta inferentzia denbora motzenak dituelako.

Metodoetako CNN sareak

*R-CNN* metodo guztietan ikusi daiteke zati garrantzitsuenetako bat CNN sare nagusiekin ezaugarri mapak lortzea dela. CNN sare hauek ere bere tamaina eta arkitektura aldatzen joan dira, metodo desberdinekin lortutako emaitzak hobetuz. Lan honetan oso erabiliak diren honako CNN sareak erabili dira:

- **Resnet [He et al., 2015]**: konputagailuen potentzia handitu ahala CNN arkitektura handiagoak exekutatzeko ahalmena lortu da. Normalean, arkitekturek geroz eta geruza gehiago eduki, orduan eta emaitza hobeak ematen zituzten. Hala ere, geruza

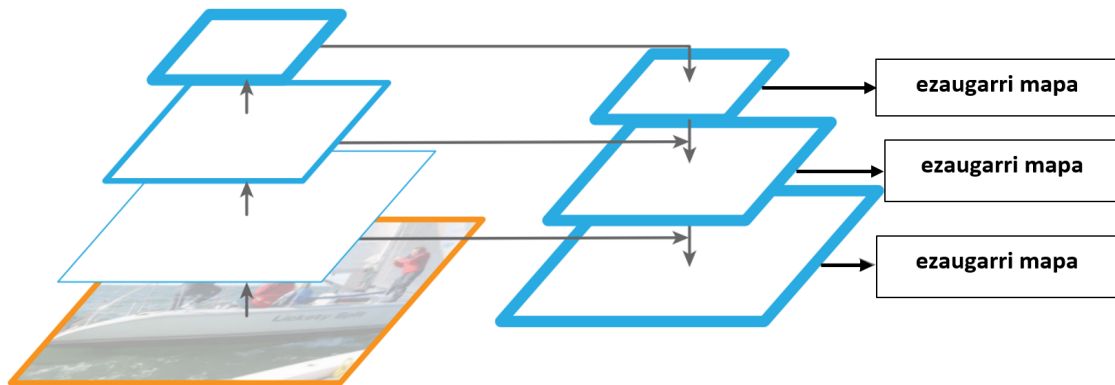
gehiegi jartzeak entrenamendua asko zailtzen zuen eta geruza gutxiagoko sareek emaitza hobekak lortzera iristen ziren. *Resnet* sareak arazo hori konpondu du *Residual Layers* deituriko geruzekin. Geruza multzo bakoitzak bidezidor bat izango du 2.12 irudian ikusi daitekeen antzera. Modu horretan, sareak geruza berri horiek emaitza hobetzeko soilik erabiliko ditu, behar ez dituenean pisuak hutsean utziz eta geruza gehigarri horiek identitate modura funtzionatuz.



2.12 Irudia: *Resnet* sareko *Residual* geruza.

Lan honetan erabilitako CNN sarea *Resnet-50* da. Sare hau 50 konboluzio geruzaz osatuta dago eta inferentzia eta entrenamendu denbora bideragarri batekin *benchmark* desberdinetan emaitza onenetatik hurbil dago.

- **Feature Pyramid Networks (FPN)** [Lin et al., 2016]: aurretik azaldutako CNN sare guztiak sarrerako irudiaren dimentsioak txikitzen joaten dira geruzetatik igaro ahala. Irudiek amaieran bereizmen txikia baina, kanal kopuru handia izango dute, posizio bakoitzaren semantika handituz. Zoritxarrez, sarean irudien dimentsioak jaisten doazen heinean, irudi originalean zeuden objektu txikiei buruzko informazioa galdu egiten da. Hori konpontzeko helburuarekin sortu dira *FPN* sareak. Sare hauek eskala desberdineko ezaugarri mapak sortzen dituzte 2.13 irudian ikusten den bezala. Lehenengo pausoa (2.13 irudiko gorako bidean), CNN sare arrunt baten antzera funtzionatzen dute, dimentsioak jaisten dira geruzetatik igaro ahala semantika handituz. Ondoren, bigarren pausoa (2.13 irudiko beheko bidean), dimentsioak handituz eta semantika maila mantenduz, eskala desberdinetako ezaugarri mapak lortzen dituzte. Beherako bidean gorako bidean erabilitako errepresentazioak ere erabiltzen dira, dimentsioen jaitsieragatik ezabatutako informazioa ez galtzeko *Resnet* sareak erabiltzen zituen *Residual* antzeko konexioak erabiliz.



**2.13 Irudia:** *Feature Pyramid Networks (FPN)* sarea.

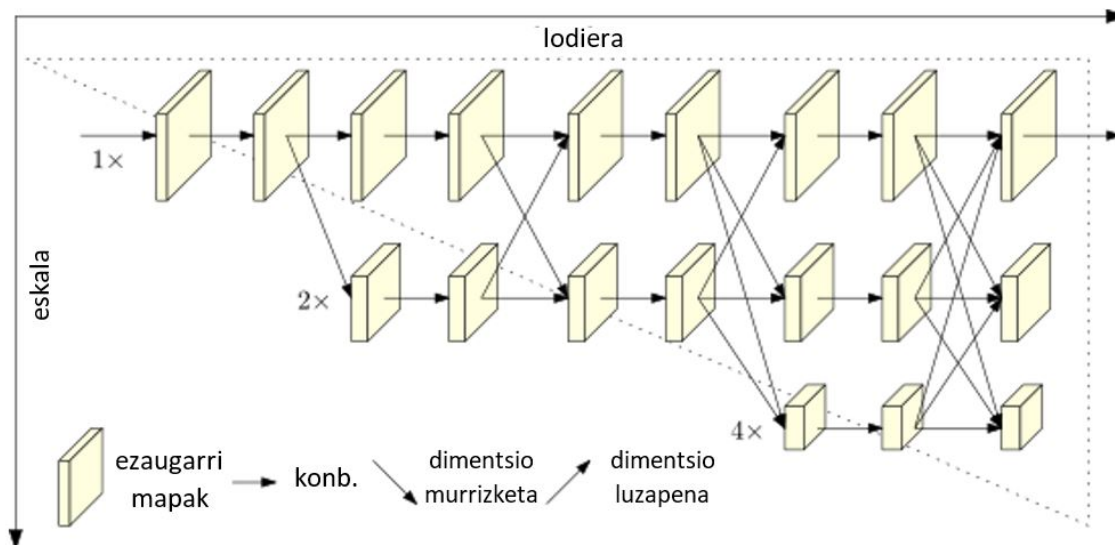
Proiektu honetan detektatu beharreko markatzaileak irudi originalaren tamainarekin konparatuz txikiak direnez, *Resnet-50* sarean oinarrituta dagoen, *FPN* moduan gorako eta beherako bideak egiten dituen eta eskala desberdineko ezaugarri mapak erabiltzen dituen CNN sare bat erabiltzen duen *Faster R-CNN* metodoa erabili da.

Segmentaziotik objektuen detekzioa

Proiektu honetan ohiko objektuen detekzioarako metodoak frogatzeaz gain, irudien instantzien segmentazioan erabilitako metodoak ere frogatu dira. [Al-Battal et al., 2021] eta [Cheng et al., 2019] lanetan erakusten da, metodoek sortutako segmentazio mapetatik, forma errazean objektu horiei dagozkien BBox-ak lortu daitezkeela. Segmentazio mapak erabiliz, ereduak objektu horien lokalizazio zehatzagoa dauka beraien detekzioa kasu askotan hobetuz. Proiektu honetan, objektuen anotazioak erabiliz, segmentazio mapak sortu dira eta irudi originaletatik segmentazio mapak sortzeko prestatutako CNN sare bat entrenatu da. Horrela, ereduari sarrera bezala irudi originalak sartuta, klase bakoitzeko segmentazio mapa bat itzultzen du.

Lan honetan segmentazioarako oso erabilia den *HRNet* sarea erabili da.

**High-Resolution Network (HRNet)** [Wang et al., 2019]: sare honek *FPN* sareen ideia berdina erabiltzen du, sarean zehar eskala desberdineko errepresentazioak erabiliz. 2.14 irudian ikusi daitekeen bezala, kasu honetan ez du gorako eta beherako bideak jarraitzen, baizik eta dimentsio originalen errepresentazioak mantentzen ditu eta bertatik dimentsio txikiagoko bide adarrak sortzen ditu.



**2.14 Irudia:** HRNet sarea.

Erresoluzio handi horren gainean konboluzioak aplikatzea eta dimentsio txikiagoko ezaugarri mapak hurrengo geruzaren sarrerarako ere erabiltzeak objektu txikiagoen segmentazioaren zehaztapena handitzen du.

### Vision Transformers

Azken urteotan *Atentzio* mekanismoetan oinarritutako *Transformer* ereduak erabiltzen hasi dira adimen artifizialeko edozein arlotan. Ikusmen artifizialaren arloan *Vision Transformers* delakoak sortu dira, CNN-ak erabili beharrean *Transformer* motako ereduak erabiltzen dituztenak. Kasu askotan, [Liu et al., 2021, Yan et al., 2022] lanetan erakutsitako bezalako ereduak aurretik sortutako edozein CNN sare baino emaitza hobeak lortzera iristen dira. Badirudi ikusmen artifizialean ere etorkizun gertuan *Transformer*-ak nagusituko direla, objektuen detekzioan, klasifikazioan eta segmentazioan lortutako emaitzak ikusirik. Hala ere, lan honetarako, erabilitako datu multzo murriztagatik, mota honetako sareak baztertu egin dira. Normalean, *Transformer* motako ereduak datu askorekin aurre-entrenatuak egoten dira eta *fine-tuning* (soilik azken geruzak arazo horretako datu multzoarekin entrenatu) bat egin ondoren, edozein datu multzorekin erabili daitezke. Lan honen kasuan, irudi polarimetrikoak erabiltzen direnez eta markatzaileak ohiko objektuekin zer ikusirik ez dutenez, ezin daiteke RGB irudiekin aurre-entrenatutako eredu bat erabili. Beraz, *Vision Transformer* moduko sareak erabiltzea baztertu da, horrelako eredu

handiak lan honetarako sortutako datu multzoarekin hutsetik entrenatzea bideragarria ez delako.





## 3. KAPITULUA

---

### Proiektuaren garapena

---

#### 3.1 Datu multzoaren eraikuntza

Proiektuko lehen lana datu multzoaren eraikuntza izan da. Horretarako, proiektuaren helburu diren markatzaileak dituzten piezen multzoari argazkiak atera zaizkie eta ondoren, ateratako argazki bakoitza anotatu da.

##### 3.1.1 Irudien lorpena

Kamera eta eszenarioa

Lan honetan erabilitako kamera polarimetrikoa 3.1 irudian ikusi daitekeen *FLIR Blackfly S BFS-U3-51S5P-C* modeloa izan da.



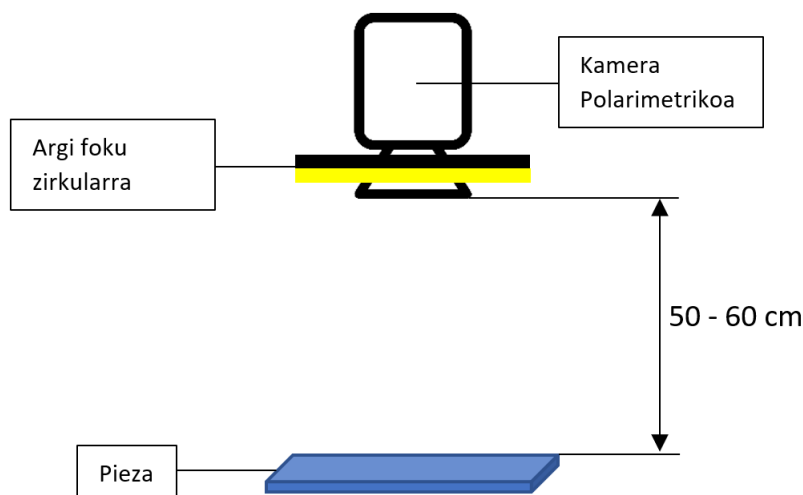
**3.1 Irudia:** *FLIR Blackfly S BFS-U3-51S5P-C* kamera polarimetrikoa.

Kamera honek pixel bakoitzeko 4 polarizazio iragazki ditu, zehazki,  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  eta  $135^\circ$ -ko angeluetako polarizazio iragazkiak eta  $1224 \times 1024$  pixeleko bereizmena dauka.

Kamera honi, makinen ikusmen aplikazioetarako bereziki sortutako *Fujinon HF12.5HA-1B* lentea jarri zaio eta berarekin ateratako argazkiek argiztapen arazorik eduki ez dezaten, LED argi-foku zirkular bat ipini da kamerako lentea inguratuz. Ipinitako argi fokua potentzia 48W-koa da.

Argazkiak ateratzeko, kamera honen API-a (*Spinnaker*) erabiltzen duen C++ lengoaiako programa bat inplementatu da. Hasierako probetan ateratako argazkiek zarata gehiegi zuten, datu multzoa osatzen zuten argazki bakoitzeko 10 desberdin atera dira, beraien arteko batez besteko irudia gordez.

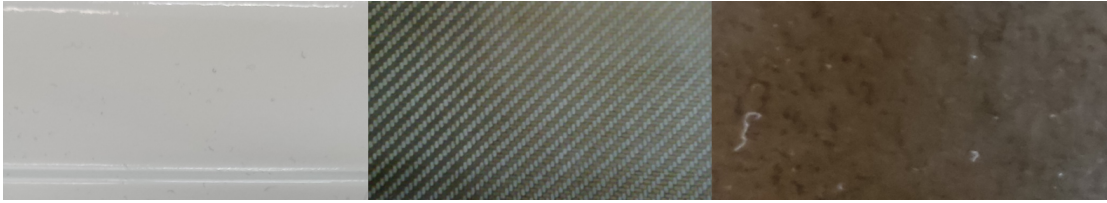
Argazkiak ateratzean, 3.2 irudian ikusi daitekeen modura, proiektuaren betebeharrengatik kamera piezatik 50-60 cm-ra ipini da, bere lentea piezarekiko ahalik eta perpendikularren egonik.



**3.2 Irudia:** Datu multzoko irudiak lortzeko eszenarioa.

Proiekturako piezak eta markatzaileak

Proiektu hau burutzeko material desberdineko 5 zuaker desberdin eduki dira. Pieza hauen material desberdinak 3.3 irudian ikusi daitezke.



**3.3 Irudia:** Piezen material desberdinak. Ezkerretik eskuinera: aluminioa, RTM zuntza eta CFRP zuntza.

Detektatu beharreko markatzaileak bi motakoak dira:

- **Zuloak:** 3.4 irudiaren ezker aldean ikusi daitezke piezaren gainean zulagailu batekin egindako zulo sinpleak direla.
- **Gurutzeak:** 3.4 irudiko eskuinaldean ikusten den moduan, gurutze formako piezaren gainazaleko erliebe mehe bat besterik ez dira.



**3.4 Irudia:** Zulo eta gurutze motako markatzaileak hurrenez hurren.

Pieza desberdinek markatzaile mota desberdinak dauzkate; batzuek zuloak soilik dituzte, beste batzuek soilik gurutzeak eta bi motatako markatzaileak dituzten piezak ere badaude.

Datu multzoko irudiak

Proiektuan erabilitako datu multzoko argazkiak bi txandatan atera dira. Lehenengo 125 irudi atera dira hasierako probak egiteko eta ondoren datu multzoa beste 229 irudirekin osatu da. Irudiak material desberdinen artean modu honetan banatu dira:

- Aluminioko piezak: 27 irudi.

- RTM zuntzeko piezak: 87 irudi.
- CFRP zuntzeko piezak: 119 irudi.
- Material desberdinetako hainbat pieza nahasturik: 121 irudi.

Material desberdinei argazki kopuru desberdina atera zaie, material bateko pieza gehiago eduki direlako bestekoak baino. Detekzioak material, markatzaile mota eta pieza guztietan ongi burutzeko, material desberdineko eta mota desberdinetako markatzaileak dituzten piezak nahastuta ageri diren 121 irudi atera dira.

Beraz, proiektuan zehar erabilitako datu multzoa 354 irudiz osatuta dago.

### 3.1.2 Anotazioak

Behin argazkiak atera ondoren, hauetan dauden markatzaile naturalak non dauden anotatu behar dira. Horretarako, doakoak diren hainbat programa probatu dira: *imglab*<sup>1</sup>, *VoTT*<sup>2</sup>, *CVAT*<sup>3</sup>, *labelimg*<sup>4</sup> eta *SuperAnnotate*<sup>5</sup>. Hala ere, irudi asko anotatu behar izan direnez, anotazioak forma errazenean eta denbora laburrenean egitea ahalbidetzen duen programa aukeratu da: **SuperAnnotate**. Lan honetarako anotazio sinpleak soilik egin behar izan direnez, *OpenCV*-rekin elkarlanean sortutako *SuperAnnotate*-ren doako bertsioa erabili da. Programa honekin mota askotako anotazioak burutu daitezke, hala nola, *Bounding Box*-ak, puntuak, zirkuluak...

Objektuen detekzioak bi modu desberdinetan (3.5 irudia) anotatu daitezke:

- **Axis-aligned Bounding Box (AABB)**: ohiko *Bounding Box*-ak, beti horizontalki eta bertikalki zuzenak direnak, ardatzekiko paraleloak izanik. Ohiko metodo guztiek objektuen identifikazio moduan erabiltzen duten *BBox* mota da.
- **Oriented Bounding Box (OBB)**: *Bounding Box* hauek ez daukate zertan X eta Y ardatzekiko zuzenak izan, angelu batez biratuta egon daitezkeelarik. Hauen kasuan, 4 puntuen koordenatu guztiekin identifikatu behar dira:

$$(x_1, y_1) (x_2, y_2) (x_3, y_3) (x_4, y_4)$$

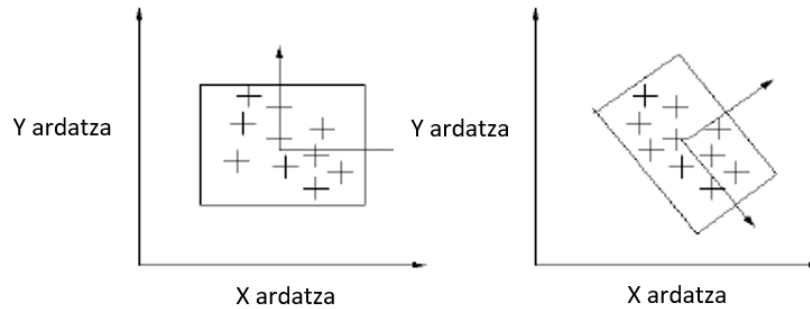
<sup>1</sup>*imglab* programa web orri honetan erabili daiteke: <https://imglab.in/>

<sup>2</sup>*VoTT* programa web orri honetan eskuratu daiteke: <https://github.com/microsoft/VoTT>

<sup>3</sup>*CVAT* programa web orri honetan eskuratu daiteke: <https://github.com/openvinotoolkit/cvat>

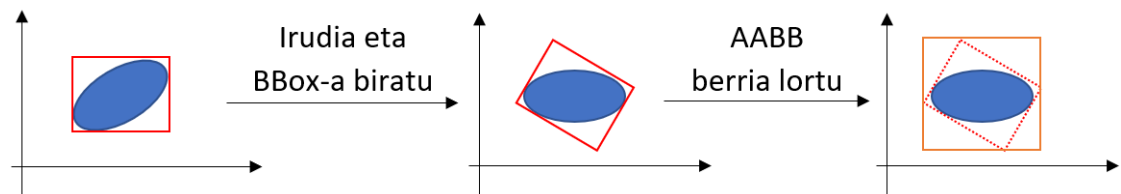
<sup>4</sup>*labelimg* programa web orri honetan eskuratu daiteke: <https://github.com/tzutalin/labelImg>

<sup>5</sup>*SuperAnnotate* programa web orri honetan eskuratu daiteke: <https://www.superannotate.com/download>



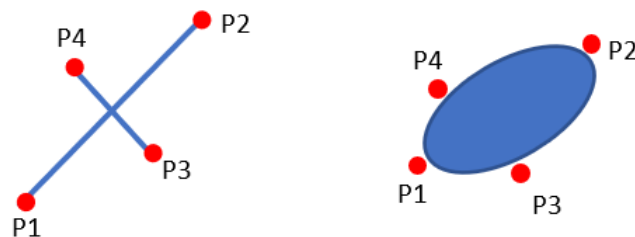
**3.5 Irudia:** *Axis-aligned Bounding Box* eta *Oriented Bounding Box* hurrenez hurren.

Proiekturako eraikitako datu multzoak irudi gehiegi ez dituzenez, irudi biraketan oinarrituta dauden eta Bounding Box-en koordinatuak ere biratzen dituzten datu gehikuntza (*data augmentation*) hainbat teknika erabili dira. AABB-ak biratzeko, 3.6 irudian ikusi daitezkeen bezala, lau erpinak biratu eta lau puntu biratu horiek barnean dituen AABB berri bat definitu behar da.



**3.6 Irudia:** AABB-en errotaziotik lortutako AABB berria.

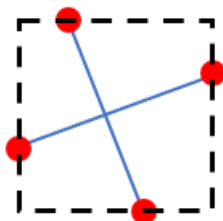
3.6 irudiko adibidean nabari da, kasu batzuetan AABB-ak anotatzen badira, ondoren biraketa aplikatzean, AABB berrien zehaztasuna ez dela batere ona. Hori konpontzeko eta ondoren segmentazioen mapen sorkuntza errazteko, 3.7 irudian ikusi daitezkeen modura, OBB-aren ardatzak binaka definitzen dituzten 4 puntu anotatu dira. Gurutze motako markatzaileen kasuan, puntu horiek gurutzeko lau erpinak dira.



**3.7 Irudia:** Gurutze eta zuloetarako anotatu diren lau puntuen adibideak.

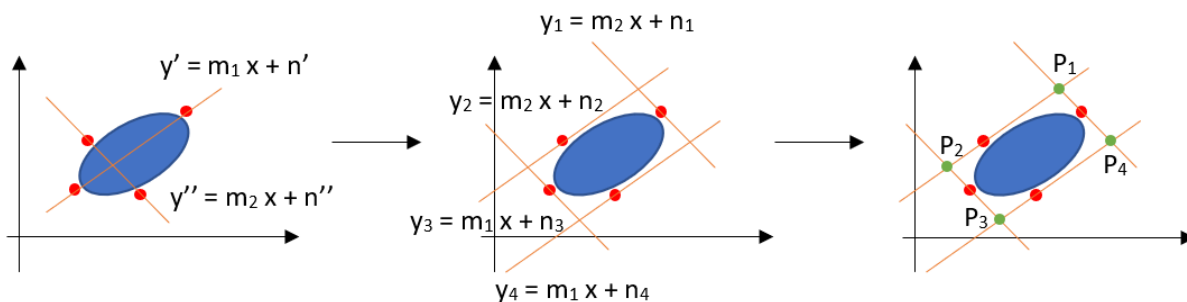
Anotatutako puntu horietatik objektuen detekzio metodoen entrenamendurako behar diren AABB-ak lortzeko, markatzaile mota bakoitzeko prozedura desberdina erabili da:

- **Gurutzeak:** anotatu diren lau puntuetatik,  $(x,y)$  balio maximoak eta minimoak erabiliz, AABB-a lortu da (3.8 irudia). Prozedura hau jarraituz, gurutzearen forma bereziagatik, anotatutako bi puntu lortutako AABB-aren erpinetan egonda ere, markatzaile osoa definitutako AABB-ren barnean egongo da. Ezaugarri hau zirkulu eta elipseekin betetzen ez denez, prozedura hau ezin da zulo motako markatzaileekin erabili.



**3.8 Irudia:** Gurutze markatzaileen kasuan, anotatutako 4 puntuetatik lortutako AABB-a.

- **Zuloak:** lehenengo, zuloen OBB-a kalkulatu da. Horretarako, 3.9 irudian azaltzen den prozedura erabili da. Lehenengo, ardatz bakoitza definitzen duten bi puntuen arteko maldak kalkulatu dira ( $m_1$  eta  $m_2$ ). Ondoren, malda horiek dituzten eta kontrako ardatzeko bi puntuetatik pasatzen diren zuzenak kalkulatu dira ( $y_1, y_2, y_3, y_4$ ). Azkenik, zuzen horien elkargune puntuak kalkulatu dira, puntu horiek OBB-a definitzen dutelarik ( $P_1, P_2, P_3, P_4$ ).



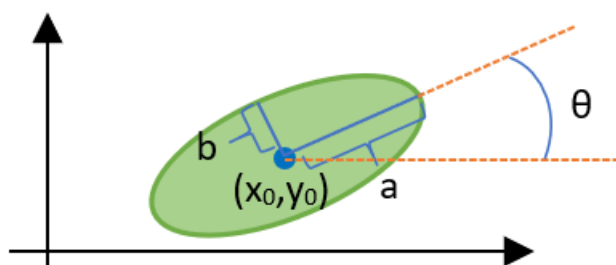
**3.9 Irudia:** Anotatutako 4 puntuetatik, OBB-a lortzeko prozesua.

OBB-a kalkulatu ondoren, zuloek elipse forma daukatenez, kalkulaturako OBB-aren puntuetatik elipsearen bi ardatzen erradioak ( $a$  eta  $b$  ardatz luzeenaren eta motzenaren erradioak izanik),  $(x_0, y_0)$  zentroa eta  $\theta$  angelua lortu dira (3.10 irudia).

Horiekin elipse horren AABB zehatzena definitzen duten bi puntuen koordenatuak 3.1 eta 3.2 formulekin kalkulatu dira.

$$x = \pm \sqrt{a^2 \cos^2 \theta + b^2 \sin^2 \theta} + x_0 \quad (3.1)$$

$$y = \pm \sqrt{a^2 \sin^2 \theta + b^2 \cos^2 \theta} + y_0 \quad (3.2)$$



**3.10 Irudia:** Zulo markatzaileen anotazioek definitutako OBB-tik, AABB-ak lortzeko, erabilitako elipsearen parametroak.

Modu honetan anotazioak eginez eta AABB-ak lortzeko prozesu horiek jarraituz, datu gehikuntza teknikekin anotatutako puntuak eta irudiak biratu arren, objektuak identifikatzen dituen ahalik eta AABB zehatzenak lortzen dira.

### 3.1.3 Segmentazio mapak

Ohiko Bounding Box-en anotazioez gain, detekziorako segmentazio mapak sortzen dituzten metodoak ere frogatzeko, egindako anotazioetatik abiatuta segmentazio mapak sortu dira. Kasu honetan bi markatzaile natural mota daudenez, bi segmentazio mapa sortu dira irudi bakoitzeko.

Irudien anotazioak egiteko erabilitako moduak lagunduta, markatzaile mota bakoitzeko segmentazio mapak sortzeko prozedura hauek erabili dira:

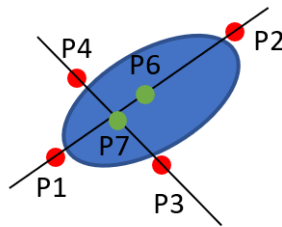
- **Gurutzeak:** anotazioak lau puntu definituz egin direnez (3.7 irudia), anotatutako lehenengo eta azkeneko bi puntuek gurutzearen bi lerro nagusiak definitzen dituzte. Segmentazio mapetan gurutzeak definitzeko, irudi huts batean (0 balioz osatuta), anotazioen puntuak erabilia gurutzearen marrak marraztu dira. Lerroak, *OpenCV*

<sup>6</sup> liburutegiaren laguntzaz, hiru pixeleko lodierarekin marraztu dira, gurutzeko le-roen pixelei leko balioa emanik.

- **Zuloak:** zuloen anotazioetako lau puntuetatik, zuloak definitzen duen elipsearen parametroak lortu dira: elipsearen zentroa, ardatzen luzera eta ardatz nagusiaren angelua.

– Elipsearen zentroa kalkulatzeko honako prozesua jarraitu da:

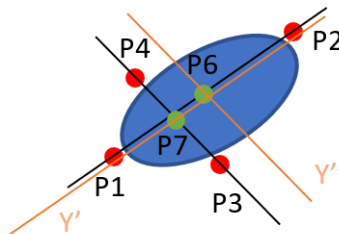
1. Ardatz bakoitza definitzen duten puntuen arteko erdiko puntuak kalkulatu.



**3.11 Irudia:** Ardatzak definitzen dituzten bi puntuen arteko erdiko puntuak.

3.11 irudian ikusi daiteke, P1 eta P2-ren arteko puntuen zentroa P6 dela eta P7 P3 eta P4-ren artekoa.

2. P1 eta P2-tik pasatzen den zuzenaren malda duen eta P7 puntutik pasatzen den zuzena kalkulatu (3.12 irudian  $Y'$ ). Forma berdinean P3 eta P4-tik pasatzen den zuzenaren malda duen eta P7tik pasatzen den zuzenarekin ( $Y''$ ).

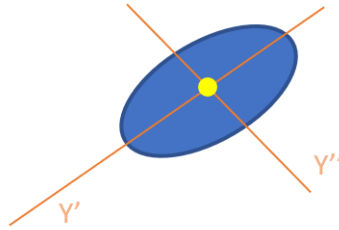


**3.12 Irudia:** Ardatzen malda bera duten eta zentroko puntuetatik pasatzen diren  $Y'$  eta  $Y''$  zuzenak.

3. Kalkulatutako bi zuzenek bat egiten duten puntua hartu elipseko zentro moduan (3.13 irudia).

<sup>6</sup>OpenCV liburutegia hemen eskuratu daiteke: <https://opencv.org/>.





**3.13 Irudia:** Bi zuzen berriek ebakitzen duten puntuak.

- Elipsearen ardatzen luzera moduan, ardatz bakoitza definitzen duten bi puntuen arteko distantziak hartu dira.
- Zulo edo elipsearen angelua ardatz nagusiaren (luzeena) zuzenaren maldari  $\arctan$  aplikatuz lortu da.

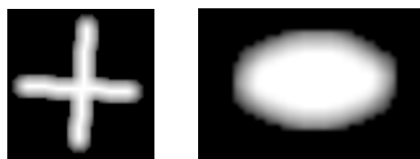
Parametro guztiak lortutakoan, *OpenCV* liburutegia erabiliz, kalkulaturako parametroak dituen elipse bete bat marraztu da, bere barneko pixelek 1eko balioa izanik.

Prozedura horiek jarraituz lortutako segmentazio mapetan, markatzaile bat dagoen pixelek bateko balioa dute eta beste guztiek berriz, zerokoa. Literaturan hainbat lanek diotenez, pixel batetik bestera aldaketa hain handiak sareari ikastea galarazten diote. Horregatik, lan askotan funtzio *Gaussian* batekin fintzen dira balioak [Wang et al., 2019]. Sortutako segmentazio mapekin berdina egiteko,  $5 \times 5$  tamainako *kernel*-a duen lausotze (*Blur*) iragazki bat aplikatu zaie sortutako segmentazio mapei (3.14 irudia).



**3.14 Irudia:** Segmentazio mapak sortu ondoren, lausotze prozesuaren eragina.

Prozesu guztia jarraitu ondoren, 3.15 irudian ikusi daitezke lortutako segmentazio mapen markatzaile mota desberdinen adibide batzuk.



**3.15 Irudia:** Gurutze eta zulo motako markatzaileei dagokien sortutako segmentazio mapak.

## 3.2 *Datu gehikuntza* teknikak

Aurreko atalean azaldu den antzera, sortutako datu multzoak ez dauzka irudi gehiegi. Entrenamenduan irudi horiek soilik erabilia *overfitting* (ikasitako ezaugarrien orokortasun falta) handia edukiko denez, datu gehikuntza teknika desberdinak erabili dira. Datu gehikuntza deritzo entrenamendu fasean, entrenatzeko dauden irudiei zenbait ausazko transformazio aplikatzeari. Horrela, irudi bakoitza iterazio guztietan ez da guztiz berdina izango.

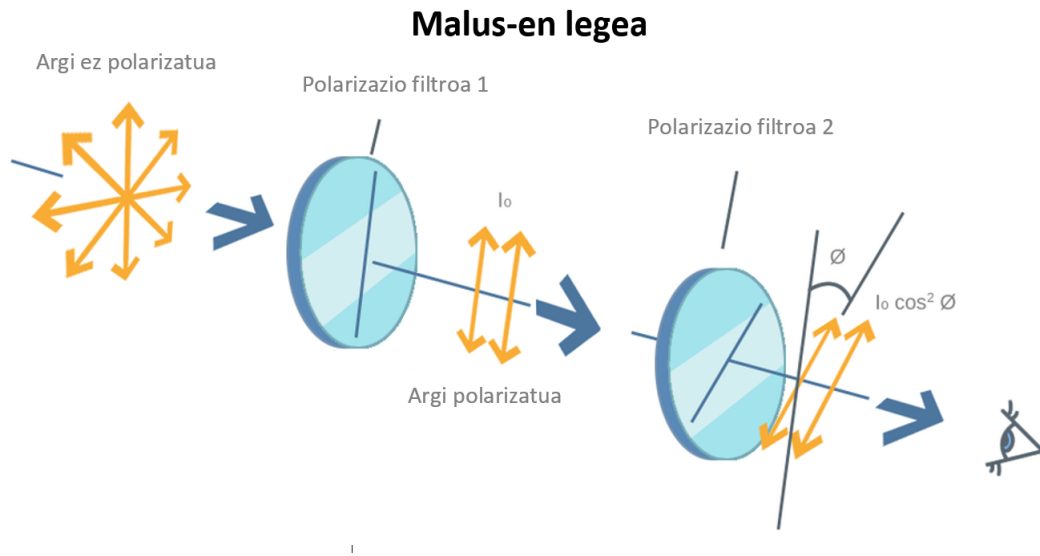
Datu gehikuntza aplikatzean, kontuan eduki behar da erabilitako irudiak ez direla ohiko RGB irudiak eta kamera polarimetrikoaren erabilpenarekin lortutako irudiei ezin zaizkiena RGB irudiei aplikatzen zaizkien ohiko transformazioak aplikatu. Hori egingo balitz, irudi berezi horien arrazonomendu fisikoa ez litzateke errespetatuko. Arrazoi honengatik, argazki mota hauei transformazio desberdinak aplikatzea proiektu honetako erronka bat izan da. Datu gehikuntzako transformazioen tekniken artean biraketa, iraulketa eta zarata aztertu dira.

### 3.2.1 Biraketa

Aurretik azaldu den antzera, kamera polarimetrikoak 4 balio itzultzen ditu pixel bakoitzeko,  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  eta  $135^\circ$ -ko polarizazio angeluei dagokien intentsitateak. Irudi hauek, RGB irudien modura ohiko biraketekin biratzen badira, beraien ezaugarri fisiko guztiak galduko dira, intentsitate balio horiek jada ez direlako angelu horiei dagozkienak izango.

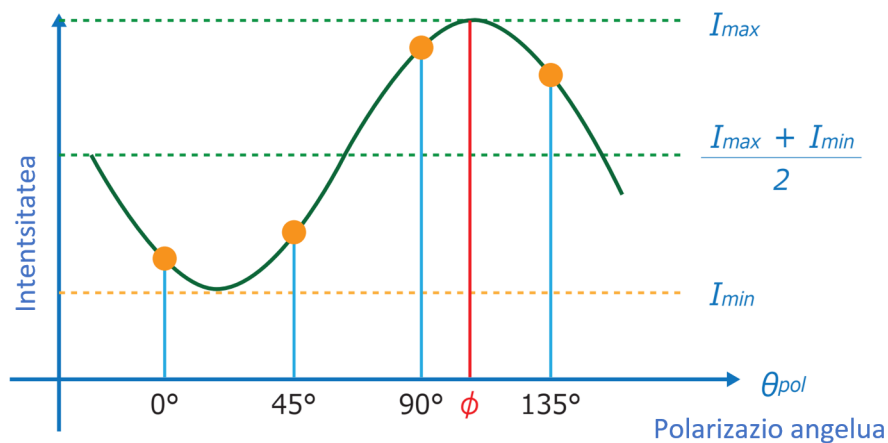
Literaturan ikertzaile batzuek sareen sarrerako irudi moduan, aurretik kalkulaturako AoP, DoP eta intentsitate orokorreko irudiak erabiltzen zituzten [Blanchon et al., 2020]. Hauen biraketarako, ohiko irudien biraketaz gain, AoP irudiaren balioei biratutako angelua kentzen zioten. Hau erabilgarria izan daiteke sareen sarrerarako irudi horiek erabiltzen badira, baina badaude kasuak non Stokes eta polarizazio angelu desberdinen intentsitate balioekin, AoP, DoP eta intentsitate orokorreko irudiekin baino emaitza hobekak lortu dituzten [Blin et al., 2019]. Hori kontuan edukita, lan honetan, polarizazio angelu desberdinen intentsitateei zuzenean biraketa aplikatu zaie, ondoren horietatik eratorritako irudi guztiak biraketa jada aplikatuta lortu daitezkeelarik.

*Malus*-en legeak dio, argi izpi polarizatua beste angelu bateko polarizazio iragazkitik pasatzean, bere intentsitate berria lortzeko, argiaren intentsitatea polarizazio angelu diferentziaren kosinuaren karratuarekin biderkatu behar dela (3.16 irudia).



**3.16 Irudia:** Malus-en legea jarraituz, argiaren intentsitatearen aldaketa bi polarizazio angeluetako iragazkiak erabiliz.

Gainazaletatik islatutako argia partzialki polarizatuta egon arren, sinplifikazio moduan polarizatu bezala erabili da. Kamera polarimetrikoak 4 angelu desberdineko iragazkiak dituela jakinda eta *Malus*-en legea jarraituz, iragazki bakoitzak emandako intentsitateak  $I_0 \cos^2(\alpha)$  funtzioko balioak izan behar dira, 3.17 irudian ikusi daitekeen bezala.



**3.17 Irudia:**  $I_0 \cos^2(\alpha)$  funtzioan, kameraren 4 iragazkien angeluen intentsitateak.

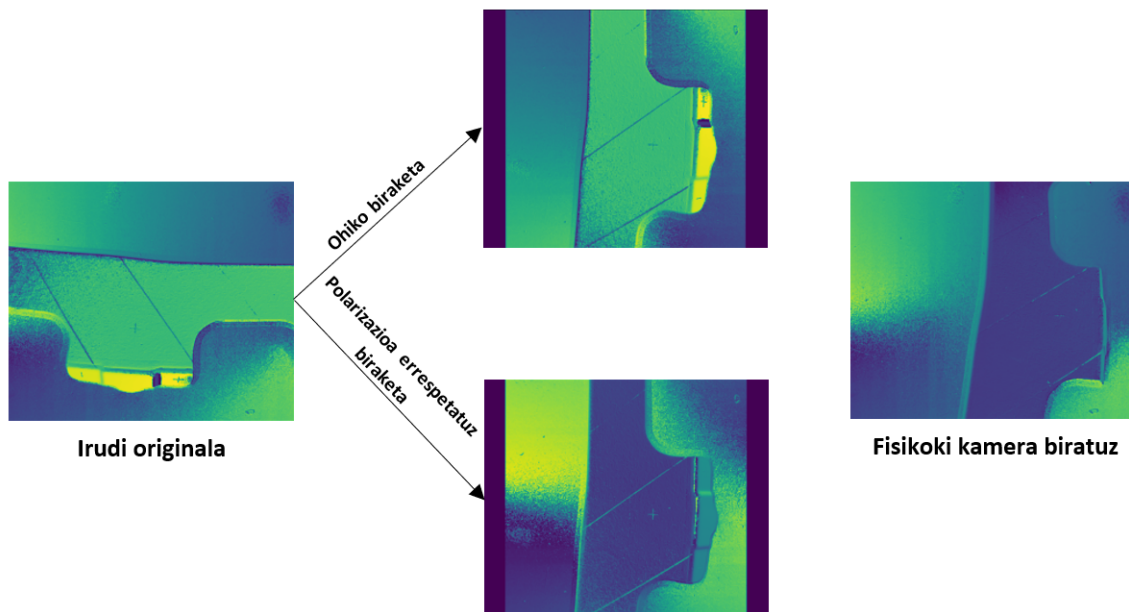
Horrez gain, irudi bat  $\delta$  angeluaz biratzea, kamera  $-\delta$  angularekin biratzearen parekoa da. Beraz, kamera biratzen bada, iragazkien angeluak ere biratu egingo dira,  $(0^\circ - \delta, 45^\circ - \delta, 90^\circ - \delta, 135^\circ - \delta)$  angeluak izanik iragazki desberdinen angelu berriak. Hori jakinda,

irudi bat  $\delta$  angeluaz biratu nahi bada, iragazkien polarizazio angelu horiei dagozkien intentsitate balioak behar dira.

Polarizazio angelu horien intentsitateak lortzeko, kamera polarimetrokoak emandako 4 angeluen ( $0^\circ, 45^\circ, 90^\circ, 135^\circ$ ) intentsitateak erabiliz,  $f(\theta) = \beta \cos^2(\theta + \alpha) + \gamma$  funtzioa optimizatu da. Prozesu horretan, funtzioko balioak kamerak itzulitako intentsitate balioetatik ahalik eta gertuen egoteko behar diren  $\beta, \alpha$  eta  $\gamma$  parametroak lortu behar dira. Funtzioa irudiaren pixel haina aldiz optimizatu behar denez, proiektu honetan optimizazioa bizkor egiteko, C++ lengoaiako, *Ceres-solver*<sup>7</sup> liburutegia erabili da.

Behin funtzioko parametroak optimizatuta, irudi bat  $\delta$  angeluarekin biratu nahi bada, irudiaren pixel bakoitzeko, bere intentsitate balioak aldatu behar dira. Horretarako,  $f(\theta - \delta)$  funtzioaren ( $0^\circ, 45^\circ, 90^\circ, 135^\circ$ ) angeluen intentsitate berriak lortu behar dira. Polarizazio intentsitateak aldatu ondoren, irudia RGB irudi baten antzera biratuko da.

3.18 irudian argi ikusi daiteke, proposatutako biraketak kameraren polarizazioaren fisika ohiko biraketak baino hobeto mantentzen duela.



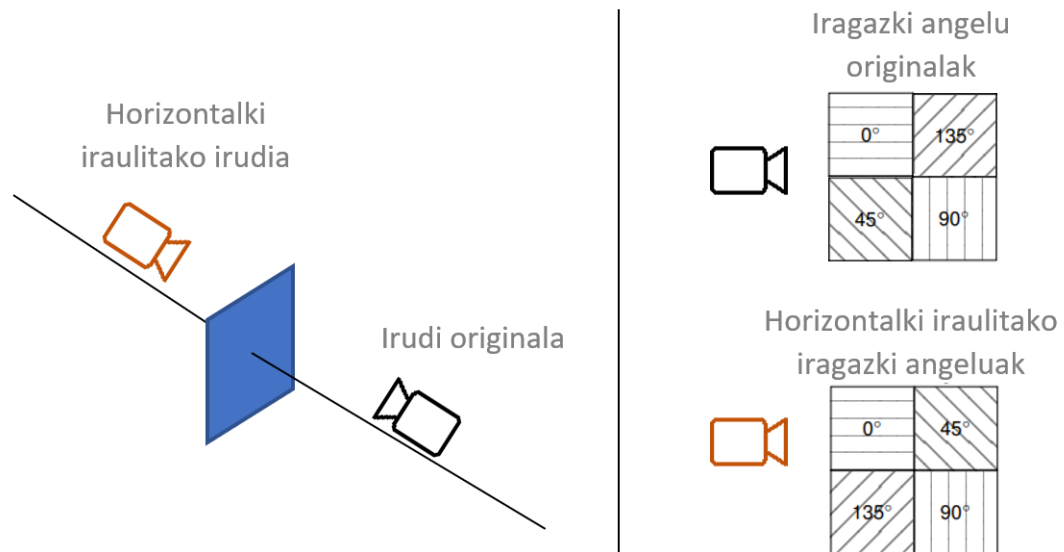
**3.18 Irudia:** Ezkerrean, irudi originalaren AoP irudia. Erdian, proposatutako eta polarizazioa errespetatzen duen metodoarekin eta ohiko metodo arruntarekin  $90^\circ$  biratuta lorturiko AoP irudia. Eskuinean, mundu errealean kamera  $-90^\circ$  biratuz lortutako AoP irudia.

<sup>7</sup>*Ceres-solver* liburutegia hemen eskuratu daiteke: <https://github.com/ceres-solver/ceres-solver>

### 3.2.2 Iraulketa

Biraketekin gertatzen den antzera, iraulketa edo *flip* transformazioak ere ezin daitezke RGB irudi baten antzera aplikatu. Literaturan AoP irudiari iraulketa transformazioak aplikatzeko, bertako angeluak  $-1$  balioaz biderkatzen dituzte [Blanchon et al., 2020]. AoP irudia alde batera utziz, polarizazio angelu desberdinen intentsitateari transformazio hau aplikatzeko, argi eduki behar da iraulketa transformazioaren esanahia.

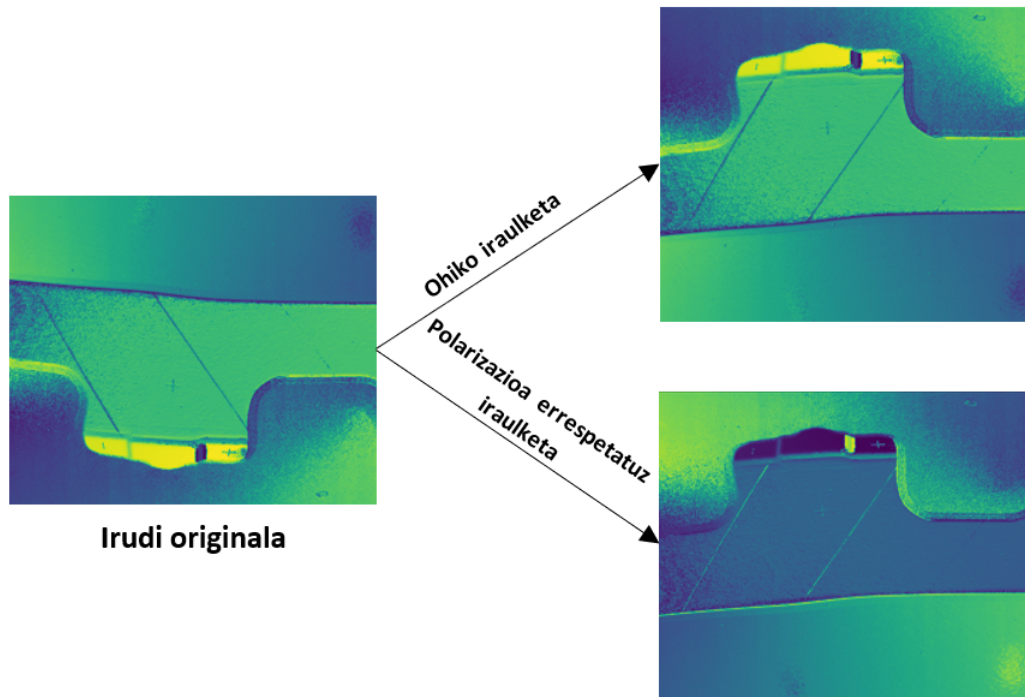
Iraulketa horizontala kamerarekin ikusten den plano berdina beste norantzan ikustearen parekoa da (3.19 irudia). Kamera polarimetrikoen kasuan pixel bakoitzeko iragazki desberdinak 3.19 irudiko ezker goialdean ikusi daitekeen antzera banatuta daude. Iraulketa transformazioa burutzen bada (beste norantzatik begiratzen bada),  $0^\circ$  eta  $90^\circ$ -ko angeluko iragazkiak ez dira aldatuko, baina  $45^\circ$  eta  $135^\circ$  angeluak beraien artean ordezkatu dira. Gauza berdina gertatuko da iraulketa bertikala egiten bada.



**3.19 Irudia:** Horizontalki *flip* bat aplikatzearen esanahia eta prozesu horretan iragazki angeluen aldaketa.

Beraz, irudi polarizatuei iraulketa aplikatzeko, iraulketa teknika ohikoaz gain,  $45^\circ$  eta  $135^\circ$  polarizazio angeluen intentsitateen balioak ordezkatu behar dira pixel bakoitzeko.

3.20 irudian ikusi daiteke, modu honetan eta ohiko teknika erabiliz irudiei *flip* bertikal teknika aplikatuz lortutako irudien desberdintasunak.

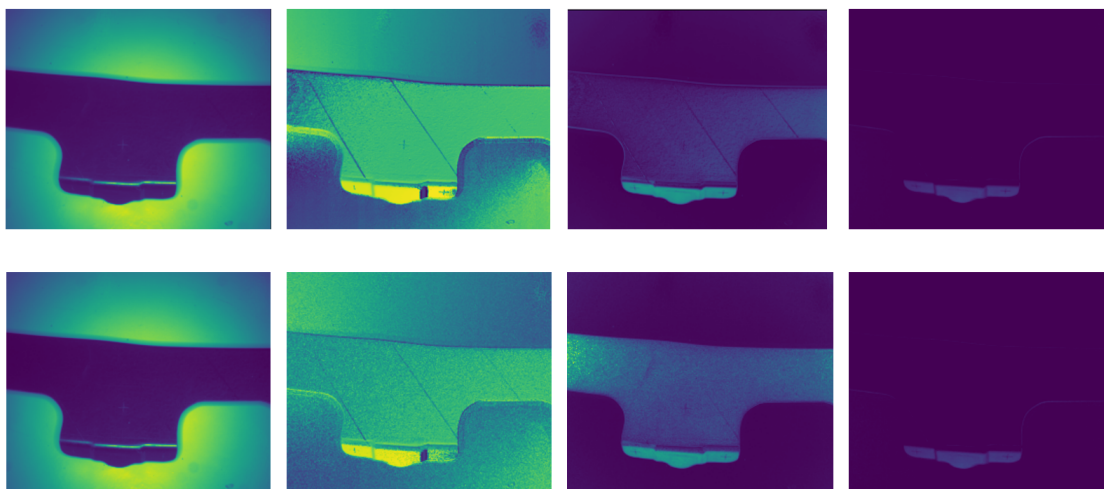


**3.20 Irudia:** Ezkerrean irudi originalaren AoP irudia. Eskuinean, polarizazioa errespetatuz proposatutako metodoaz bertikalki iruliz eta iraulketa metodo arrunta aplikatuz lortutako AoP irudiak.

### 3.2.3 Zarata

Arazo motaren arabera, irudiei zarata sartzeak ikasketa orokortzean lagundu dezakeela ikusi da [Moreno-Barea et al., 2018]. Horregatik, kasu honetan datu gehikuntza moduko beste teknika baten antzera, ausazko zarata sartu zaie irudiei. Hori egiteko  $N(0,3)$  banaketa normaleko ausazko balioak gehitu zaizkie pixel bakoitzeko polarizazio angeluen intentsitateen balioei.

Intentsitate hauek  $[0,255]$  tartera normalizatuta daudenez, zarata oso urria sartu da, al-daketak oso txikiak izango direlarik. 3.21 irudian ikusi daiteke, irudi original eta zarata sartutako iruditik eratorritako irudien desberdintasuna. Polarizazio angelu desberdinen intentsitateetan, distirarik gabeko irudian bezala, aldaketarik ia ez da sumatzen. Horietatik eratorritako AoP eta DoP-etan berriz, zarataren ondorioak nabarmenagoak dira.



**3.21 Irudia:** Ezkerretik eskuinera: distirarik gabeko irudia, AoP, DoP eta  $S_2$  irudiak. Goiko irudiak zaratarik sartu gabe lortu dira, behekoak berriz, irudi originalari zarata sartuta.

### 3.3 Detekzioa

Datu multzoa eraikitakoan, oinarri teorikoetan objektuak detektatzeko bi metodo aukeratuak (*Faster R-CNN* eta *HRNet*) entrenamenduak eta inferentziak burutu dira. Detekzioaren prozesu osoa *Python* lengoaia eta neurona-saretarako oso erabilia den *Pytorch*<sup>8</sup> liburutegia erabiliz burutu da.

#### 3.3.1 Datu multzoaren banaketa

Prozesu osorako sortutako datu multzoa modu honetan zatitu da:

- **Train multzoa:** irudi guztien %64 ditu (227 irudi). Irudi gehien dituen multzoa da, entrenamendurako erabiltzen dena delako. Irudi multzo honen inferentziaren emaitzekin aplikatzen da ikaskuntza-prozesua, metodoko ereduaren pisuak optimizatzeko.
- **Validation multzoa:** irudi guztien %16 ditu (56 irudi). Irudi gutxien duen multzoa da. *Validation* multzoa metodo desberdinen hiperparametroak eta entrenamendu prozesuan, prozesu osoko sareen pisu onenak zein aroetakoak (*epoch*-etakoak) diren zehazteko erabiltzen da.

<sup>8</sup>*Pytorch* liburutegia honako estekan eskuratu daiteke: <https://pytorch.org/>.

- **Test multzoa:** irudi guztien %20 ditu (71 irudi). Irudi multzo hau entrenamenduan ez da erabiltzen. Behin metodoko sareen entrenamendua burutu ondoren, irudi multzo hau sarearen emaitzen kalitatea neurtzeko erabiltzen da.

Multzoak ausazko modu batean aukeratu dira, beti ere material desberdinen piezen irudiak modu justu batean multzo guztien artean banatuz.

### 3.3.2 Azpiegitura

Entrenamendu prozesu guztiak *Amazon Web Service (AWS)*-ko *SageMaker Studio*<sup>9</sup> zerbitzuan burutu dira. Zerbitzu honetan beraien zerbitzarietan exekututzen diren *Jupyter-notebook* moduko fitxategiak erabili daitezke. *AWS*-k bakoitzaren beharren araberrako hardware desberdineko instantziak eskaintzen ditu. Kasu honetan, neurona-sareak entrenatu behar izan direnez, GPU-en erabilera beharrezkoa izan da eta horregatik, *g4dn.xlarge* instantzia aukeratu da. Instantzia honen ezaugarriak hauek dira:

- NVIDIA Tesla T4 GPU bat
- 4 nukleoko Intel Cascade Lake prozesadorea
- 16GB RAM
- 125GB SSD NVMe

Proiektu hau *IDEKO Zentro Teknologikoan* burutu denez, azpiegitura hauek beraiek hornituak izan dira.

### 3.3.3 Faster R-CNN

Oinarri teorikoetan azaldu den modua, *Faster R-CNN* metodoak, sarrera bezala irudiak emanda, definitutako klase bakoitzeko irudian detektatutako *BB*ox-ak itzultzen ditu. Metodo honen entrenamendurako, datu multzoko *train* irudi multzoa eta horietan dauden markatzaileei dagokien *AABB*-ak erabili dira. Sortutako datu multzoaren anotazio moduan *OBB*-en ardatzen puntuak gordetzen direnez, puntu horietatik dagokion *AABB*-a lortu dira.

---

<sup>9</sup>*Amazon Web Service*-eko zerbitzu guztiak honako estekan eskuratu daitezke: <https://aws.amazon.com/es/>.



Proiektu honetan, *Torchvision* liburutegiak *Pytorch*-erako eskaintzen duen, ereduaren implementazioa hartu da, zehazki, proiektu honetako objektuak txikiak direnez, *Resnet-50*-an oinarritutako *FPN* moduko sarea oinarri duen *Faster R-CNN*-a <sup>10</sup> erabili da. Metodo honetan erabilitako ereduak guztira 41 milioi parametro dituzte.

Entrenamendu garaian, *COCO* [Lin et al., 2014] datu multzotik aurre-entrenatutako pisuak hartu dira hasierako puntu bezala. Hala ere, proiektu honetako irudi eta objektuak datu multzo horretako irudi eta objektuekin zer ikusirik ez dutenez, entrenamendu garaian sare guztietako geruza guztiak entrenatu behar izan dira. Sare honi sarrera moduan irudi originalen bereizmen osoko irudiak eman zaizkio, hau da,  $1224 \times 1024$  bereizmenekoak.

Entrenamendu garaian parametro hauek aukeratu dira:

- Optimizatzailea: *Adam*. Beste optimizatzaileen moduko emaitzak lortzen ditu, abiadura azkarragoa izanez eta bere parametroen balio desberdinen aurrean zurruntasun gehiago erakutsiz [Kingma and Ba, 2014].
- Ikaskuntza-tasa (*Learning rate*): hainbat baliorekin probak egin ondoren,  $1e - 4$  balioa izan da onena.
- Sorta-tamaina (*Batch Size*): GPU-aren memoriak ahalbidetu duen balio handiena 8-koa izan da.
- Aro (*Epoch*) kopurua: 300.
- Galera (*loss*) funtzioa: bi galera balio erabiltzen ditu, *RPN* ereduarena eta detekzio orokorrarena. Bi balioak 3.3 funtzioarekin kalkulatu dira, *RPN* ereduak eta *Faster R-CNN* metodo orokorrak lortzen dituzten BBox-en koordinatu eta klase bakoitzaren probabilitate desberdinekin (*RPN* ereduaren emaitzetan, klaseak objekturik egon edo ez egon dira).

$$L(\{p_i\}, \{t_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*) \quad (3.3)$$

Funtzio horretan BBox bakoitza zein klasekoa den probabilitatearen ( $\{p_i\}$  lortutakoa eta  $\{p_i^*\}$  benetakoa) galera balioa ( $L_{cls}$ ) 3.4 funtzioarekin kalkulatu da, klaseen arteko *log-loss*-a kalkulatu.

<sup>10</sup>Implementazioa honako estekan eskuratu daiteke: [https://pytorch.org/vision/main/generated/torchvision.models.detection.fasterrcnn\\_resnet50\\_fpn.html](https://pytorch.org/vision/main/generated/torchvision.models.detection.fasterrcnn_resnet50_fpn.html).

$$L_{cls}(p_i, p_i^*) = -p_i^* \log p_i - (1 - p_i^*) \log(1 - p_i) \quad (3.4)$$

Beste zatia BBox-en koordinatuei dagokie.  $t_i$  eta  $t_i^*$  bakoitza BBox-aren koordinatu multzoa dela jakinda ( $t_i$  metodoak estimatutako BBox-aren koordinatuak eta  $t_i^*$  berriz, anotatutako BBox-arenak), beraien arteko diferentziaren gehikuntzarekin kalkulatzen da BBox-en koordinatuen erregresioari dagokion errorea ( $L_{reg}$ ), 3.5 funtzioa jarraituz.

$$L_{reg}(t_i, t_i^*) = \sum_{u=\{x_{min}, y_{min}, x_{max}, y_{max}\}} |t_i[u] - t_i^*[u]| \quad (3.5)$$

Azkenik,  $N_{cls}$  eta  $N_{reg}$  balioak normalizatzeko erabiltzen dira.  $\lambda$  balioa berriz, klaseen galera balioen eta koordinatuen galera balioaren arteko oreka bilatzeko erabiltzen da, gomendatuena 10 balioa izanik [Ren et al., 2015].

#### Anchor Box-en aukeraketa

*Faster R-CNN* metodoko *RPN* eredia sarrerako iruditik lortutako ezaugarri maparen zati desberdinetan aplikatzen da. Horietan, intereseko eskualdeak identifikatzeko, aurrez definitutako *Anchor Box* izenez ezagutzen diren tamaina eta aspektu ratio desberdineko BBox-ak daude. Oro har, tamaina handiko objektuak identifikatzeko *Anchor Box*-en tamaina handi bat ezarri beharko da, baina lan honetan markatzaileen tamaina txikia denez, *Anchor Box* tamaina txikiak frogatu dira.

125 irudiz osatutako datu multzo txiki batekin *Anchor Box* tamaina desberdineko (4 pxeletatik 512 pxeletara) multzoekin entrenatu da.

3.1 taulan ikusi daiteke, multzo desberdinen artean ez dela desberdintasun handirik egon. Hala ere, emaitza onenak (16, 32, 64, 128, 256) tamaina multzoarekin lortu dira eta horregatik, tamaina multzo hori erabili da hurrengo pausu guztietan.

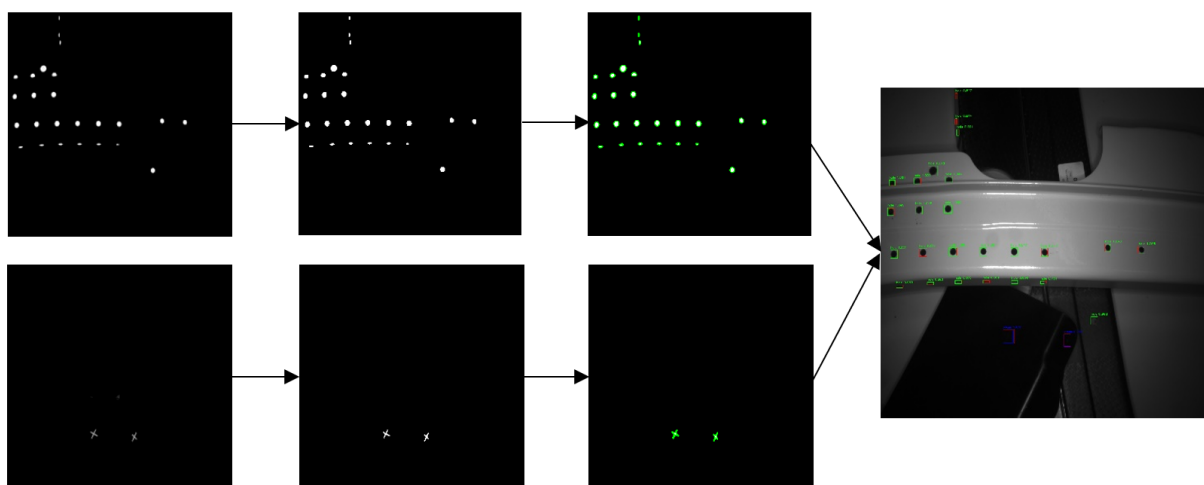
<b>Anchor Box tamainak</b>	<b>mAP (zuloak, gurutzeak)</b>	<b>mAP<sub>50</sub></b>	<b>mAP<sub>75</sub></b>
(4, 8, 16, 32, 64)	(0.6057, 0.6772)	0.9316	0.7569
(8, 16, 32, 64, 128)	(0.6028, 0.6895)	0.9171	0.7468
(16, 32, 64, 128, 256)	<b>(0.6132, 0.7104)</b>	<b>0.9388</b>	<b>0.7664</b>
(32, 64, 128, 256, 512)	(0.5076, 0.6780)	0.9140	0.7645

**3.1 Taula:** *Anchor Box* tamaina desberdineko multzoekin *Faster R-CNN* metodoa entrenatuta lortutako emaitzak.

### 3.3.4 HRNet

*HRNet* ereduak sarrera moduan sartutako irudietatik, horiei dagokien klaseko bakoitzeko segmentazio mapa bat itzultzen du. Itzulitako segmentazio mapa hauen tamaina sarrerako irudien tamaina baino txikiagoa denez, sarrerako irudien bereizmen berdinerara eskalatu dira.

Mapa eskalatu horietatik BBox-ak lortzeko prozesua *OpenCV* liburutegia erabiliz burutu da. Horretarako, segmentazio mapa horiek balio finko batekin binarizatu dira (segmentazio mapak  $[0,1]$  tarteko balioak dituzte, objektuen detekzioaren konfiantza maila moduan erabili daitezkeenak) eta *Close* eragiketa morfologiko bat aplikatu zaie markatzaileen inguruko zarata ezabatzeko. Azkenik, irudi binarizatu horietan siluetak bilatu dira. Silueta bakoitza markatzaile baten moduan hartu da, ingurada hori definitzen duen BBox-a detekzioaren emaitza izanik. BBox horien konfiantza mailaren balio moduan, segmentazio mapako BBox horretako pixel balio handiena ezarri da. Prozesuaren pausuak 3.22 irudian ikusi daitezke.



**3.22 Irudia:** Segmentazio mapetatik BBox-ak lortzeko prozesua. Goran zulo eta behean gurutze motako markatzaileak. Ezkerretik eskuinera, *HRNet* ereduak itzulitako segmentazio mapak, mapa binarizatuari *Close* eragiketa morfologikoa aplikatuta lortutako mapak, horietan aurkitutako siluetak berdez ageri diren irudiak eta azkenik, segmentazio mapetatik lortutako BBox-ak.

Proiektuan bere egile ofizialek *Pytorch* liburutegirako egindako *HRNet* ereduaren inplementazioa <sup>11</sup> erabili da, zehazki, 32 geruza dituen *HRNet\_w32* eredu. Eredu honek 28,5 milioi parametro ditu.

<sup>11</sup>*HRNet* sarearen inplementazioa honako estekan eskuratu daiteke: <https://github.com/leoxiaobin/deep-high-resolution-net.pytorch>.

Entrenamenduan, ereduaren hasierako pisuak *COCO* datu multzoarekin entrenatuak izan dira, baina aurretik aipatu bezala, proiektuan erabilitako datu multzoa eta *COCO* datu multzoaren desberdintasun handiagatik ereduaren geruza guztiak entrenatu dira. Sarre-  
ra moduan, altuera eta zabalera berdineko irudiak izatea behartzen duenez, proiektuko  $1224 \times 1024$ -ko irudiak,  $1024 \times 1024$  bereizmenera eskalatu dira.

Entrenamendu garaian parametro hauek aukeratu dira:

- Optimizatzailea: *Adam*.
- Ikaskuntza-tasa: *Faster R-CNN*-aren moduan, hainbat baliorekin probak egin ondoren,  $1e - 4$  balioak eman ditu emaitza onenak.
- Sorta-tamaina: kasu honetan eredu honek *Faster R-CNN*-ko ereduak baino parametro gutxiago eduki arren, GPU-aren memoriak ezin izan du 8 irudiko sorta-tamaina izan, horregatik, 4-koa erabili da.
- Aro kopurua: 300.
- Galera funtzioa: ereduak itzultitako eta anotazioetatik sortutako segmentazio mapen arteko *Mean Square Error (MSE)*. 3.6 funtzioan ikusi daiteke MSE-a nola kalkulatu den.

$$L_{MSE}(m, m^*) = \frac{1}{C * H * W} \sum_{i=0}^C \sum_{l=0}^H \sum_{t=0}^W (m_i(l, t) - m_i^*(l, t))^2 \quad (3.6)$$

Bertan,  $m$  ereduak lortutako segmentazio mapak dira eta  $m^*$  anotazioetatik lortutako segmentazio mapak.  $C$  klase kopurua da eta  $H$  eta  $W$  segmentazio mapen altuera eta zabalera. Proiektu honetan 2 klase eta  $1024 \times 1024$  tamainako segmentazio mapak erabili dira.

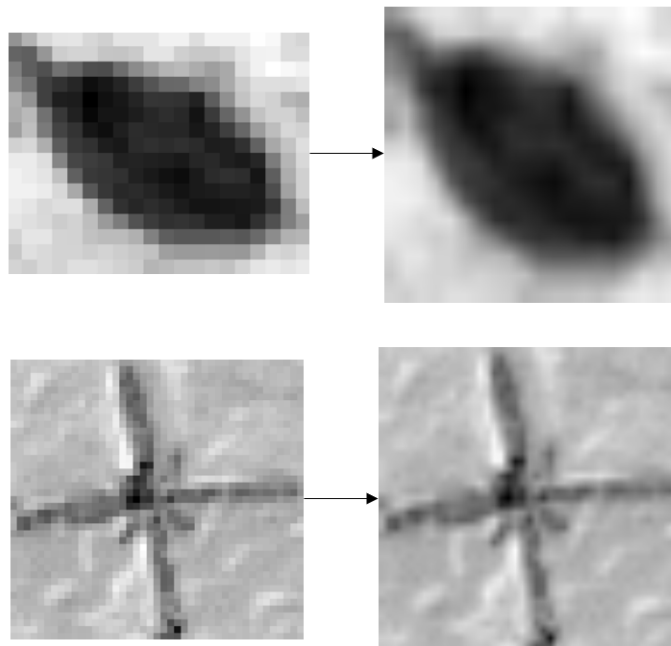
### 3.4 Markatzaileen zentroen zehaztapena

*HRNet* ereduko BBox detekzioetatik abiatuta, detektatutako markatzaile bakoitzaren zentroaren detekzio ahalik eta zehatzena egiten saiatu da. Prozesu hau konputagailu bidezko teknika klasikoagoekin egiten da, alde batetik zehaztasun handia (milimetroen azpitik) birlatzen denez, Deep Learning eredu bat entrenatzeko irudi bakoitzean markatzaile guztien anotazio oso zehatzak beharko liratekeelako. Beste alde batetik, produktu komertzialetan

zehaztasuna bilatzen ari denean, Machine Learning ereduaren ausazkotasuna eta zurruntasun falta ez dira batere ona, industria arloan programak egoera guztietan modu berean joka dezan bilatzen delako (errepikakortasuna). Bi arrazoi horiengatik, lehen detekzioa Deep Learning ereduarekin egitea erabaki da eta ondoren, emaitza horiek erabiliz, ahalik eta zehaztasun gehienez markatzaileen zentroa forma klasikoago batean bilatzea.

Detekzio zehatzago horretarako *Python* lengoaiako *OpenCV* liburutegiko ikusmen artifizialeko prozedurak erabili dira.

Prozesu honetan pixelen azpiko zehaztasuna bilatzen denez (sub-pixelak), detektatutako BBox-ei dagokien irudiaren zatia, tamaina handiago batera linealki eskalatu da. Lan honetan egindako probetan, irudiak eskalatu diren tamaina finko hori 1024x1024-koa izan da (3.23 irudia). Hori egitean, irudi eskalatu horretako kontrastea hobetzeko, bere balio tartea [0,255]-era normalizatu da.

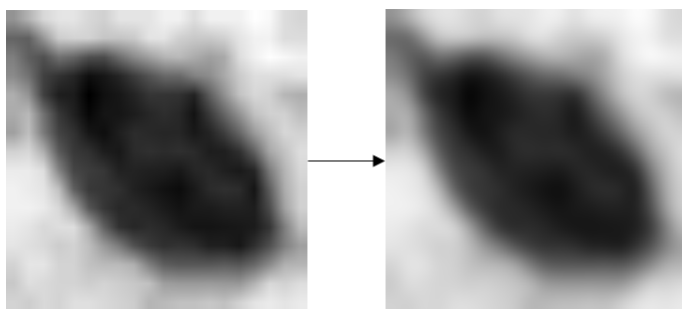


**3.23 Irudia:** Detektatutako markatzaileari dagokion irudi zatia eta irudi berdina linealki eskalatuta 1024 x 1024 tamainara.

*HRNet* ereduaren detekzioetan bi markatzaile motak bereizten direnez (markatzaile bakoitza klase bat da), markatzaileari dagokion irudi zatia eskalatu ondoren, markatzaileen zentroaren detekzioa forma desberdinean egin da.

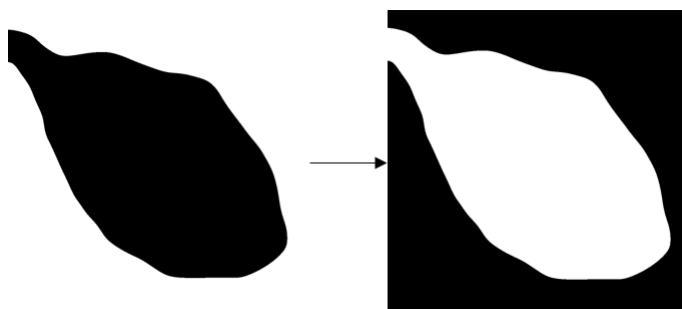
### 3.4.1 Zuloak

Zuloen zentroaren detekziorako lehenik eta behin irudia lausotu egin da *Gaussian Blur* iragazki bat aplikatuz. 3.24 irudian ikusi daitekeen moduan, iragazki hau aplikatzean, zuloaren ertzak fintzen dira, zuloaren ertzetik hurbil dauden pixelen balio desberdinak orekatzen direlarik.



**3.24 Irudia:** *Gaussian Blur* iragazkia aplikatu aurretik eta ondoren.

Lausotze prozesua burututakoan, *Otsu* metodoa [Otsu, 1979] erabiliz irudia binarizatu da. Horrela, irudiak [0-255] arteko balioak eduki beharrean, soilik 0 edo 255 balioak izango ditu. Zuloa definitzen duten pixelek beti 255 balioa eta irudiko hondoak 0 balioa izatea erabaki denez (irudien argi edo piezen material desberdinengatik baliteke kasu guztietan horrela ez izatea), irudi binarizatuko 4 erpinetako 50x50 tamainako zatietan dagoen balioen batez bestekoa kalkulatu da. Balio hori altuegia bada, hondoak 255 balioa duela esan nahiko du eta beraz, kasu horretan 3.25 irudian ikusi daitekeen modura, balioak alderantzikatu dira.



**3.25 Irudia:** Irudi binarizatua eta zuloari dagokion pixelen balioek 255 balioa izan dezaten, balioak alderantzikatuta.

Irudi binarizatutik abiatuta bertako silueta edo ingeradak bilatu dira. Aurkitutako silue-

ta horietako bat zulo markatzaileari dagokiona izango denez, azalera handieneko silueta hartu eta elipse batean doitzen da. Kasu batzuetan, irudia kalitate onekoa bada, iruditik zuloaren silueta egokia lortzen da, baina beste batzuetan irudi kalitatea ez denean hain ona binarizazioan ez delako zuloa guztiz ondo agertzen, ezin daiteke zuloari dagokion ingerada aurkitu. Desberdintasun hau [3.26](#) irudian argi ikusten da.

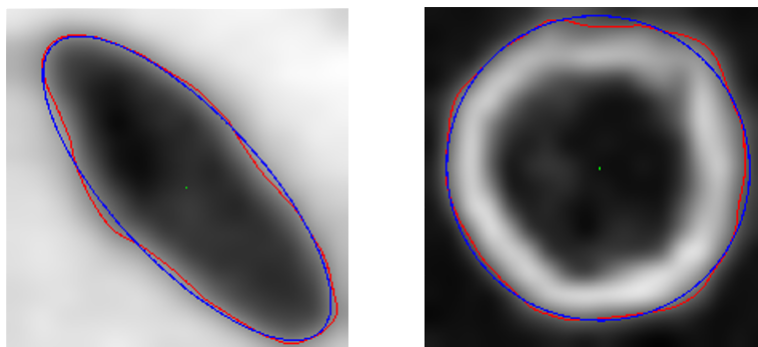


**3.26 Irudia:** Kalitate oneko eta kalitatea txarreko irudi binarizatuak hurrenez urren.

Kalitate txarreko irudietatik markatzaileen zentroak zehazki lortu ezin daitezkeenez, irudi horiek baztertzen dira. Irudiak baztertzeko bi irizpide erabiltzen dira zuloen kasuan:

- Aurkitutako siluetetatik handienak, irudi osoaren azaleraren  $\frac{1}{5}$ -a baino gutxiago izatea. Lan honetan markatzaileen irudiak 1024x1024 tamainara eskalatu direnez, 200.000 *pixel*<sup>2</sup>-an zehaztu da azalera minimoaren balio-muga hori.
- Siluetan doitzen den elipsearen eta siluetako puntuen arteko batez besteko distantzia maximo bat ezarri da, zulo itxurarik ez daukaten siluetak baztertuz. Lan honetan, batez besteko balio maximo horri 40 pixeleko balioa eman zaio.

Irizpide horiek jarraituz, baztertzen ez diren irudietan, detektatutako elipsearen zentroaren koordinatuak irudi originaleko koordinatuetara transformatuz lortuko dira zulo motako markatzaileen zentroak. [3.27](#) irudian ikusi daitezke, markatzaileetatik lortutako zentroak.



**3.27 Irudia:** Zuloen zentroen detekzioen adibideak. Gorriz detektatutako zuloaren silueta, urdinez doitutako elipsea eta berdez elipsearen eta beraz, markatzailearen zentroa.

### 3.4.2 Gurutzeak

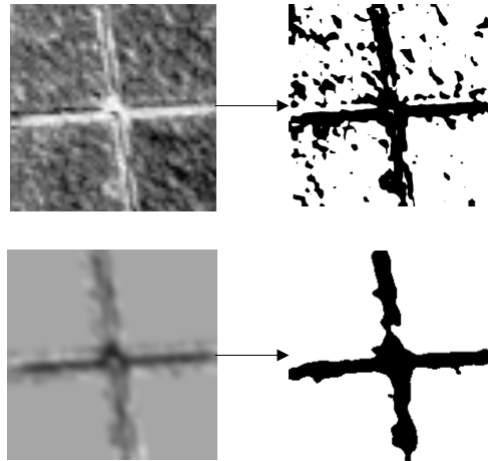
Gurutzeen zentroen detekziorako, *HRNet* ereduak itzultako segmentazio mapak erabili dira hondoko zarata ezabatzeko. Horretarako, detekzioari dagokion segmentazio maparen zatia hartu da, bertako markatzailea dilatatu da eta irudi originalaren gainean maskara bezala erabili da. Irudi originalei maskara aplikatzean, maskara horretako pixelen balioa ez da aldatu, baina maskaratik kanpo dauden pixelei irudi originaleko balioen mediana esleitu zaie (gurutze markatzaileak dauden irudietan, gurutzeari dagokien pixelak hondoko pixelak baino gutxiago direlako erabili da mediana). Maskara aplikatu ondoren, zuloen kasuan bezala, irudia lausotu eta *Otsu* metodoarekin binarizatu da, gurutzeari dagokien pixelek 255 balioa izan dezaten, behar denean balioak alderantzizkatuz. Segmentazio mapak maskara moduan erabilia eta erabili gabe lortutako irudi binarizatuak 3.28 irudian ikusi daitezke. Irudi horietan argi geratzen da hondoko zarata ezabatzen laguntzen dutela maskarak.

Maskara aplikatu, lausotu eta binarizatu ondoren, irudiko siluetak bilatzen dira. Aurkitutako siluetetatik irudi osoaren luzeraren eta zabaleraren %90-a gutxienez zeharkatzen ez duten siluetak ezabatzen dira, silueta horiek ezin izan daitezkeelako gurutze markatzaileenak izan.

3.28 irudiko irudi binarizatuetan argi ikusten da, detekzio zehatza egiteko gurutzea osatzen duten bi lerroak zabalegiak direla, lerro horiek findu behar direlarik. Horretarako, *Zhang-Suen* mehetze algoritmoa erabili da [Zhang and Suen, 1984]. Algoritmo honekin lortutako irudien adibideak 3.29 irudian ikusi daitezke.

Ondoren, gurutzea osatzen duten lerroak detektatzeko *Hough* transformazioa erabili da



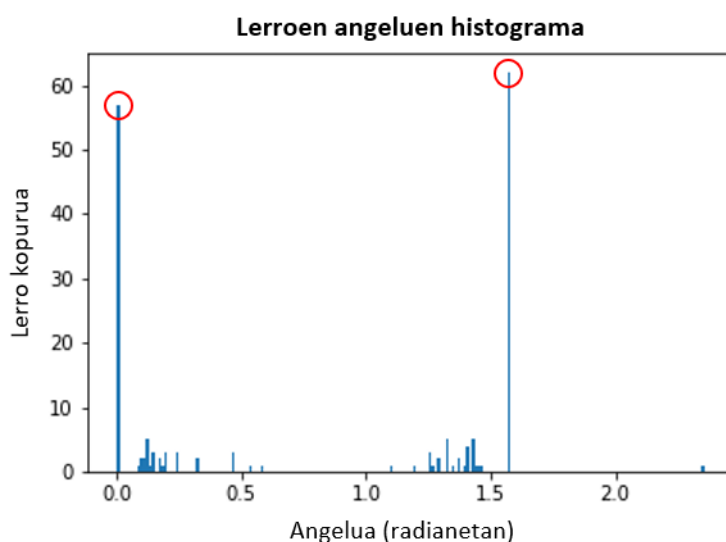


**3.28 Irudia:** Beheran maskara aplikatuz lortutako irudi binarizatua eta goran berriz, maskarik aplikatu gabe. Bi kasuetan irudi binarizatuak 0 eta 255 balioak alderantzikatu gabe daude oraindik.



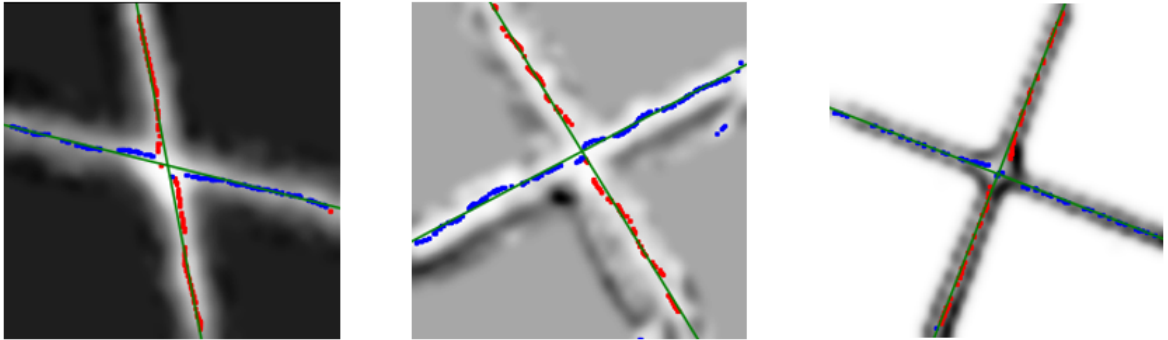
**3.29 Irudia:** Irudi binarizatuari *Zhang-Suen* mehetze algoritmoa aplikatuz lortutako irudiak.

[Hough, 1962]. Transformazio honek irudi binarizatu bat emanda bertan detektatutako lerro zerrenda itzultzen du. Lerro kopuru handiegia detektatzen duenez, lerro horien angeluekin histograma bat eraiki da. Histograma horretan, lerro gehien dituzten bi angelu desberdinak hartu eta angelu horretako lerroak definitzen duten puntuak gordetzen dira. 3.30 irudian ikusi daiteke, irudi batean aurkitutako lerroen angeluen histograma eta aukeratutako bi tontorrak.



**3.30 Irudia:** Irudi batetik detektatutako lerroen angeluen histograma. Gorriz lerro gehien dituzten angeluak adierazten dira (0 eta  $\frac{\pi}{2}$ ).

Aukeratutako angelu bakoitzeko lerroen puntuekin erregresio lineala burutzen da, gurutzearen bi lerro nagusiak lortzeko. Erregresioan, puntu guztietatik oso urruti dauden *outlier*-ak kontuan ez hartzeko, horretarako prestatuta dagoen *Huber* motako erregresio lineala burutu da [Huber, 1964]. Azkenik, bi lerro horien ebakitzaille puntuak irudi originaleko koordinatuetara transformatuta izango da markatzailearen zentroa. 3.31 irudian ikusi daitezke prozesu honekin lortutako emaitzak.



**3.31 Irudia:** Gurutzearen zentron detekzioaren prozesua aplikatzean lortutako emaitzen adibideak. Gorriz eta urdinez gurutzeko lerro bakoitzeko erabilitako puntuak eta berdez estimatutako gurutzearen bi lerroak.

Prozesu osoan zehar, zuloen prozeduran bezala, irudi batzuen kalitate txarragatik detekzio zehatzik burutu ezin daitekeenez, baztertu egin behar dira. Gurutzeen kasuan, baztertzeko irizpide hauek erabili dira:

- Irudiei maskarak aplikatu, lausotu eta binarizatu ondoren, irudiaren altueraren eta zabaleraren %90-a zeharkatzen duen siluetarik ez badago, irudia baztertzen da. Honek esan nahiko du, irudi binarizatuan ez dagoela siluetarik gurutze markatzailea definitzen duenik.
- Iruditik lerroak bilatzeko *Hough* transformazioa burutzean lerrorik aurkitzen ez bada, irudia baztertzen da.
- Angelu nagusiak aukeratzean eta angelu hori duten lerroen puntuekin erregresio lineala burutzean, bertan estimatutako lerroak puntuekiko batez besteko errore maximo bat ezarri da. Errorea maximo hori baina altuagoa bada, irudia baztertzen da.
- Gurutzearen lerro nagusi bat soilik aurkitu bada, irudia baztertzen da.



## 4. KAPITULUA

---

### Esperimentuak eta emaitzak

---

#### 4.1 Esperimentazioa

##### 4.1.1 Detekzio metodoak

*HRNet* eta *Faster R-CNN* bi metodo desberdinen portaera definitzeko hainbat entrenamendu egin dira. Lehenengo probak 125 irudiz osatutako datu multzoarekin burutu dira eta azken probak berriz, datu multzo osoarekin.

Lehenengo, aurreko 3.2 kapituluan azaldutako datu gehikuntza teknika desberdinekin entrenatuz, ea emaitzak hobetzen dituzten ikusi da. Entrenamendu desberdinetan erabilitako hurrengo datu gehikuntza teknika konbinazioak frogatu dira:

- Irudien ausazko biraketa, polarizazioa ere kontuan edukita. Iterazio bakoitzean irudia biratzen da, ausazko  $[0, 2\pi]$  tarteko biraketa-angeluarekin.
- Irudien ausazko bertikal eta horizontal iraulketak, polarizazio ere kontuan edukita. %50-eko probabilitatearekin bertikalki iraultzen da. Horrez gain, %50-eko probabilitatearekin horizontalki ere iraultzen da.
- Irudian ausazko zarata sartzea.
- Ausazko biraketa eta zarata.
- Ausazko biraketa eta iraulketa.

- Ausazko biraketa, iraulketa eta zarata.

Azken hiru konbinazioak datu gehikuntza teknika desberdinak beraien artean nola funtzionatzen duten ikusteko erabili dira. Horrez gain, aipatu behar da irudiei datu gehikuntza teknikak soilik entrenamendu prozesuan aplikatu zaizkiela eta ez baliozkotze edo test prozesuetan.

Horrez gain, irudi polarimetrikoen *Deep Learning*-erako erabileran, sareen zein sarrera irudi erabili definitu behar da. [Blin et al., 2019] lanean polarizazio angelu desberdinen intentsitate balioak hartu dituzte metodoen sarrera gisa eta [Blanchon et al., 2020] lanean berriz, *AoP*, *DoP* eta intentsitate orokorrak erabiltzen dituzte. Arazo desberdinetarako sarrera desberdinak erabiltzen direnez, sarrera konbinaketa desberdinekin bi metodoak entrenatu dira. Ondoren, entrenamendu hauen emaitzak aztertu dira arazo honetarako zein sarrera irudi diren onenak definitzeko. Sareak hiru *channel*-eko zabalera edukitzeko presatua daudela kontuan harturik, 3 sarrera iruditako konbinazioak egin dira. Hauek izan dira erabilitako sarrera irudi konbinazioak:

- **$I_0, I_{45}, I_{90}$** : kamera polarimetrikoak gordetzen dituen polarizazio angelu desberdinen intentsitateak.
- **$S_0, S_1, S_2$** : intentsitateetatik lortutako *Stokes* balioak.
- ***AoP, DoP, GlareRed***: pixel bakoitzeko polarizazio angelua, polarizazio maila eta iruditik distirak ezabatu ondoren lortutako argi intentsitate orokorra.

Lortutako detekzioetatik *Faster R-CNN* metodoaren kasuan, zein BBox-en konfiantza maila muga jarri behar den definitu behar da. *HRNet*-aren kasuan, BBox-ak lortzeko prozesuan zein konfiantza mailarekin irudia binarizatu behar den definitu behar da. Horretarako, bi kasuetan probak egin dira [0.1, 0.3, 0.5, 0.7, 0.9, 0.99] konfiantza balioekin eta bakoitzean egiazko positibo (TP), positibo faltsu (FP) eta negatibo faltsuen (FN) oreka onena zein konfiantza mailarekin lortzen den ikusi da.

Azkenik, aurreko probak egindakoan, metodo desberdinek kamera polarimetrikoak eskaintako onurak eta polarizazioren fisika bere detekzioetan baliatzen ari diren ikusteko bi proba egin dira:

- Biraketa eta iraulketa datu gehikuntza teknikak aplikatzean, polarizazio kontuan hartzen duten metodoak aplikatu beharrean, *RGB* edo irudi monokromatiko arruntei aplikatzen zaizkien prozedurak aplikatzea.

- Kamera polarimetrikoa erabili beharrean kamera monokromatikoa erabiltzearen simulazioa egitea. Horretarako, sarrera irudi bezala polarizazio angelu bakarreko intentsitateen irudia soilik erabili da, bere balioak emanez sarrerako irudiko 3 *channel*-ei.

#### 4.1.2 Detekzio zehatza

Markatzaile desberdinen zentroa bilatzeko prozedurak definitu ondoren, zein irudiren gainean exekutatu diren definitu behar da. Aurretik azaldu bezala, kamera polarimetrikoa erabiltzean 10 irudi desberdin lortzen dira, detekzio zehatz honetan zein irudik laguntzen duen aztertu behar delarik.

Detekzio zehatz hauen kalitatea neurtzeko bi neurketa egin behar dira:

- Zehatz detektatu ahal izan diren portzentaia. Prozeduretan, kalitate txarreko irudiak baztertzeko direnez, baztertze eta kalkulatu daitezkeen irudien ratioa aztertu behar da.
- Irudietatik markatzaileen zentroa modu egokian detektatzean, detekzio horren zehaztasuna kalkulatu behar da.

Bigarren neurketa egiteak konplexutasun handia dauka. Zehaztasun handi horiek neurtzeko *Laser Tracker* edo *Coordinate-measuring machine (CMM)* moduko makinak behar dira eta makina horiek alokatzea kostu handia dauka. Horregatik, esperimentazio hori burutzeko denbora askoko planifikazioa eta prestaketa behar izaten denez, zehaztasunaren neurketa ezin izan da proiektu honetan burutu.

Lehenengo neurketa berriz, burutu da, irudi desberdinen gainean prozedurak aplikatuz. Hori egitean, sarrera irudi bakoitzari aplikatzean lortutako emaitzak batu dira, irudi guztiak erabiliz markatzaileen zentroa ea zein portzentaiekin aurkitu ahal izan den ikusteko.

#### 4.1.3 Integrazioa

Integrazio prozesuarekin hasteko, proiektu honetan proposatutako sistema *Jetson Nano* mikroordenagailu batean frogatu da. *Jetson Nano*-a adimen artifizialeko ereduen inferentziak burutzeko, *NVIDIA* konpainiak plazaratutako mikroordenagailu bat da (4.1 irudia). Deep Learning ereduen inferentzia denbora azkartzeko, mikroordenagailu honek bere barnean *NVIDIA* GPU bat dauka. Bere tamaina eta kostu txikiagatik, industria arloan oso erabilia da adimen artifizialeko prozesuak industrializatzeko.



**4.1 Irudia:** *Jetson Nano* mikroordenagailua.

*Jetson Nano*-an *HRNet* eredu entrenatuak daukan inferentzia denbora eta erabilitako memoria kantitatea neurtu da. Horretarako, zenbaki errealen bi zehaztasun frogatu dira:

- *Floating-point 32* (FP32): zenbaki errealak gordetzeko 32 bit edo 4 byte erabiltzen ditu.
- *Floating-point 16* (FP16): zenbaki errealak gordetzeko 16 bit edo 2 byte erabiltzen ditu.

FP16 moduan, zenbakiak izan dezaketen zehaztasuna jaisten denez, emaitzetan izandako galerak ere neurtu dira.

Horrez gain, ereduaren entrenamenduan erabilitako *Pytorch* liburutegiarekin probak egiteaz gain, *NVIDIA*-k garatutako *TensorRT*<sup>1</sup> liburutegia ere erabili da. Liburutegi hau, *NVIDIA* GPU-ekin inferentziak ahalik eta azkarren burutzeko prestatuta dago.

*HRNet* ereduarekin inferentzia denborak neurtzeaz gain, lortutako segmentazio mapetatik BBox-ak lortzeko prozesuaren eta detekzio zehatzaren denborak ere neurtu dira.

---

<sup>1</sup>*TensorRT* liburutegia esteka honetan lor daiteke: <https://github.com/NVIDIA/TensorRT>



## 4.2 Emaitzak

### 4.2.1 Detekzio metodoak

Datu gehikuntza teknikak

125 irudiko datu multzoarekin egindako probetan, sarrera moduan polarizazio angelu desberdinen intentsitateak erabiliz ( $I_0, I_{45}, I_{90}$ ) eta datu gehikuntza teknika desberdinekin entrenatuta, 4.1 taulan ikusi daitezke *Faster R-CNN* metodoaren emaitzak. Horrez gain, datu multzo berdinarekin proba berdina *HRNet* ereduarekin eginda lortutako emaitzak 4.2 taulan ikusi daitezke.

Datu gehikuntza teknikak	klase bakoitzeko mAP (zuloak, gurutzeak)	mAP <sub>50</sub>	mAP <sub>75</sub>
<b>Erabili gabe</b>	(0.6695, 0.5393)	0.8953	0.6467
<b>Biraketa</b>	(0.6741, <b>0.7060</b> )	0.9497	0.8608
<b>Iraulketak</b>	(0.7054, 0.6653)	0.9232	<b>0.8788</b>
<b>Zarata</b>	(0.6709, 0.5091)	0.9084	0.7015
<b>Iraulketak + Zarata</b>	(0.6782, 0.6469)	0.9127	0.8463
<b>Iraulketak + Biraketa</b>	( <b>0.7155</b> , 0.6840)	<b>0.9516</b>	0.8145
<b>Iraulketak + Zarata + Biraketa</b>	(0.6634, 0.6630)	0.9217	0.8664

**4.1 Taula:** Datu gehikuntza teknika desberdinekin *Faster R-CNN* metodoa erabiliz lortutako emaitzak.

Datu gehikuntza teknikak	klase bakoitzeko mAP (zuloak, gurutzeak)	mAP <sub>50</sub>	mAP <sub>75</sub>
<b>Erabili gabe</b>	(0.4988, 0.4865)	0.7882	0.5369
<b>Biraketa</b>	( <b>0.5471</b> , 0.4653)	0.8844	0.6142
<b>Iraulketak</b>	(0.4825, 0.4559)	0.8131	0.5104
<b>Zarata</b>	(0.3878, 0.3817)	0.7253	0.3499
<b>Iraulketak + Zarata</b>	(0.4214, 0.4187)	0.7456	0.5274
<b>Iraulketak + Biraketa</b>	(0.532, <b>0.5731</b> )	<b>0.9422</b>	<b>0.6156</b>
<b>Iraulketak + Zarata + Biraketa</b>	(0.4512, 0.4479)	0.8065	0.5145

**4.2 Taula:** Datu gehikuntza teknika desberdinekin *HRNet* metodoa erabiliz lortutako emaitzak.

Kontuan eduki behar da, hasierako detekzio hauetan, interes gehien duen balioa  $mAP_{50}$  dela, BBox-en prezisioa beharrean, markatzaileak detektatzea garrantzitsuagoa delako.

Emaitzetan ikusi daiteke bi metodoetan iraulketak eta biraketak laguntzen dutela, teknikarik erabili baino emaitza hobekien lortzen dituztelako, baina ez da hori gertatzen zaratarekin. *Faster R-CNN*-an, zarata beste teknikekin konbinatuz, hau erabili gabe baino emaitza okerragoak lortzen dira eta *HRNet*-aren kasuan, zarata entrenamenduan erabiltzeak teknikarik erabili gabe lortutako emaitzak okertzen ditu. Hori kontuan hartuta, hurrengo pausuetako entrenamenduetan iraulketa eta biraketa teknikak soilik erabiliko dira datu gehikuntza gisa.

Sarrera irudi desberdinak

Datu multzo osoa erabilia, metodoei sarrera moduan irudi desberdinak emanda entrenatu dira. 4.3 taulan ikusi daitezke lortutako emaitzak.

Sarrerako irudiak	$I_0, I_{45}, I_{90}$	$S_0, S_1, S_2$	AoP, DoP, GlareRed
<i>Faster R-CNN</i>	0.9508	0.9532	0.9515
<i>HRNet</i>	<b>0.9558</b>	0.9074	0.9426

**4.3 Taula:** Sarrera irudi desberdinekin *HRNet* eta *Faster R-CNN* metodoak erabiliz lortutako  $mAP_{50}$  emaitzak.

Bertan ikusi daiteke, sarrera irudi desberdinak erabiliz emaitzak ez direla asko aldatzen, denen arteko diferentzia maximoa %5-koa izanik. Hala ere, *HRNet* metodoan emaitza onenak polarizazio angelu desberdinen intentsitateekin lortzen dira. Emaitza horietatik gertu polarizazio angelua, maila eta distirarik gabeko irudiak sarturik lortutako emaitzak daude eta txarrenak *Stokes* balioekin lortutakoak dira. *Faster R-CNN*-aren kasuan sarrera guztiekin ia emaitza berdina lortu dira. Metodoen artean ere ikusi daiteke, emaitza oso antzekoak izan arren, *HRNet* ereduak lortu duela emaitza onena.

Sarrera moduan irudi desberdinak sartuz dauden desberdintasunak eta bi metodo desberdinek emandako balioak dituzten desberdintasunak ez direnez handiak, bakoitzaren doitasun eta estaldura orokorrak kalkulatu dira. Hainbat proba egin ondoren, TP, FP eta FN-en arteko oreka kontuan harturik, *Faster R-CNN* metodoak itzulitako BBox-ak hartzeko konfiantza maila egokiena 0.7-koa dela aztertu da. *HRNet*-aren kasuan berriz, itzulitako segmentazio mapetatik BBox-ak lortzeko muga balio egokiena 0.5 izan da. Metodo bakoitzerako egokiena den konfiantza maila erabilia lortutako doitasun eta estaldura balioak 4.4 taulan agertzen dira.

Sarrerako irudiak	$I_0, I_{45}, I_{90}$		$S_0, S_1, S_2$		<b>AoP, DoP, GlareRed</b>	
Neurketa	Doit.	Estal.	Doit.	Estal.	Doit.	Estal.
<b><i>Faster R-CNN</i></b> (0.7)	0.96	0.939	0.94	0.968	0.948	0.972
<b><i>HRNet</i></b> (0.5)	<b>0.977</b>	0.965	0.969	0.96	0.952	<b>0.978</b>

**4.4 Taula:** Sarrera irudi desberdinekin *HRNet* eta *Faster R-CNN* metodoak erabiliz lortutako doitasun eta estaldura emaitzak.

Doitasun eta estaldura emaitzetan  $mAP_{50}$ -etan ikusi den antzera, sarrera irudien artean desberdintasuna oso txikia da. Kasu honetan, angelu desberdinen intentsitateekin lortzen dira emaitza onenak, polarizazio angelu, maila eta distira gabeko irudiarekin lortutakoetatik oso gertu egonik. Metodoen artean aurretik ikusi den antzera desberdintasuna %1-2koa da, *HRNet*-arekin emaitza hobeak lortzen direlarik.

#### Polarizazio erabilera

Beste edozein kamera monokromatiko arrunt batekin lortutako irudiak erabili beharrean, kamera polarimetrikoa erabiltzeak izandako eragina kuantifikatzeko hainbat proba egin dira. 4.5 eta 4.6 tauletan ikusi daiteke sarrera irudi desberdinak erabilia, polarizazioaren fisika errespetatzen duten datu gehikuntza biraketa eta iraulketa teknikak eta ohiko irudiei aplikatutako biraketa eta iraulketa teknikekin entrenatuta lortutako emaitzak.

Sarrerako irudiak	$I_0, I_{45}, I_{90}$	$S_0, S_1, S_2$	<b>AoP, DoP, GlareRed</b>
<b><i>Faster R-CNN</i></b> (Polarizazio <i>Data Aug.</i> )	0.9508	0.9532	0.9515
<b><i>Faster R-CNN</i></b> ( <i>Data Aug.</i> Orokorra)	0.9409	0.9226	0.9346
<b><i>HRNet</i></b> (Polarizazio <i>Data Aug.</i> )	0.9558	0.9074	0.9426
<b><i>HRNet</i></b> ( <i>Data Aug.</i> Orokorra)	0.9383	0.9365	0.9167

**4.5 Taula:** Sarrera irudi desberdinekin eta datu gehikuntza moduan, polarizazioaren fisika errespetatzen duten teknikak eta teknika arruntekin *HRNet* eta *Faster R-CNN* metodoak entrenatuz lortutako  $mAP_{50}$  emaitzak.

Sarrerako irudiak	$I_0, I_{45}, I_{90}$		$S_0, S_1, S_2$		AoP, DoP, GlareRed	
Neurketa	Doit.	Estal.	Doit.	Estal.	Doit.	Estal.
<b>Faster R-CNN</b> (0.7) (Polarizazio <i>Data Aug.</i> )	0.96	0.939	0.94	0.968	0.948	0.972
<b>Faster R-CNN</b> (0.7) ( <i>Data Aug.</i> Orokorra)	0.933	0.966	0.9	0.942	0.936	0.959
<b>HRNet</b> (0.5) (Polarizazio <i>Data Aug.</i> )	0.977	0.965	0.969	0.96	0.952	0.978
<b>HRNet</b> (0.5) ( <i>Data Aug.</i> Orokorra)	0.97	0.97	0.956	0.972	0.936	0.972

**4.6 Taula:** Sarrera irudi desberdinekin eta datu gehikuntza moduan, polarizazioaren fisika errespetatzen duten teknikak eta teknika arruntekin *HRNet* eta *Faster R-CNN* metodoak entrenatuz lortutako doitasun eta estaldura emaitzak.

Ikusi daitekeen antzera, emaitzak nahiko berdinak dira. Polarizazioa errespetatzen duten datu gehikuntza teknikak ez dute entrenamenduaren emaitzetan hobekuntza handirik eragiten.

Emaitza horiek baieztatzeko  $I_0$  intentsitate irudia soilik erabiliz entrenatu dira bi metodoak. Honekin, kamera monokromatiko arrunt bat simulatu nahi izan da, guztiz erreala ez izan arren, beste polarizazio angeluen intentsitate balioak erabiltzen ari den erakutsiko duelarik. 4.7 eta 4.8 tauletan ageri dira proba horien emaitzak.

Sarrerako irudiak	$I_0, I_{45}, I_{90}$	$I_0$
<b>Faster R-CNN</b>	0.9508	0.9307
<b>HRNet</b>	0.9558	0.9192

**4.7 Taula:** Sarrera moduan soilik  $I_0$  erabilia eta angelu desberdinen balio guztiak erabilia lortutako  $mAP_{50}$  emaitzak.

Sarrerako irudiak	$I_0, I_{45}, I_{90}$		$I_0$	
Neurketa	Doit.	Estal.	Doit.	Estal.
<b>Faster R-CNN</b> (0.7)	0.96	0.939	0.943	0.96
<b>HRNet</b> (0.5)	0.977	0.965	0.947	0.97

**4.8 Taula:** Sarrera moduan soilik  $I_0$  erabilia eta angelu desberdinen balio guztiak erabilia lortutako doitasun eta estaldura emaitzak.

Emaitza horiek aurretik ateratako ondorioak baieztatzen dituzte.  $I_0$  soilik erabilia emaitzak antzekoak dira gehienetan %2 inguru okertuz. Hala ere, ondorioztatu daiteke, bi

metodoek polarizazioaren fisika ez direla gehiegi erabiltzen ari detekzioak lortzeko. Honen arrazoia, problema errazegia edota metodoak modu honetako informazioa erabiltzeko prestatuta ez egotea izan daiteke.

Errekurtso eta denbora

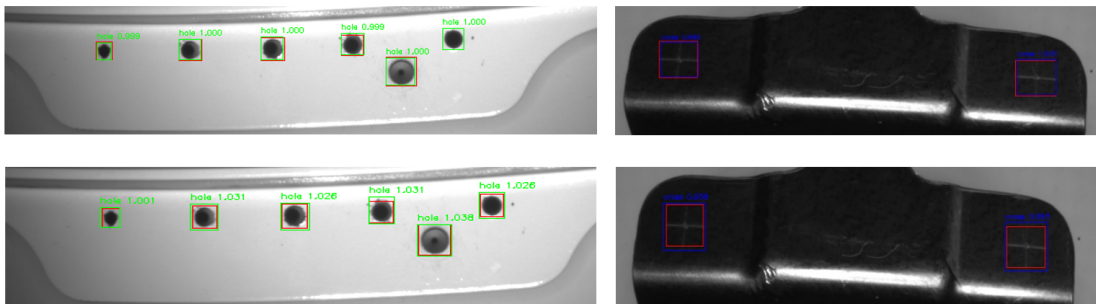
Bi metodoen entrenamendu denbora, memoria erabilera (VRAM GPU-a erabiltzen bada, RAM bestela) eta irudi bakoitzeko inferentzia denbora desberdina izan da. 4.9 taulan ikusi daiteke informazio hori.

Metodoa	Entrenamendua (GPU)		Inferentzia			
			CPU		GPU	
	Denb.	Mem.	Denb.	Mem.	Denb.	Memoria
<i>Faster R-CNN</i>	10h 24min	10,41GB	7s	700MB	90ms	500MB
<i>HRNet</i>	14h 43min	12,61GB	8s	400MB	140ms <sup>2</sup>	340MB

**4.9 Taula:** Metodo desberdinen entrenamendu eta inferentzia denbora eta memoria erabilpena.

Detekzio adibideak

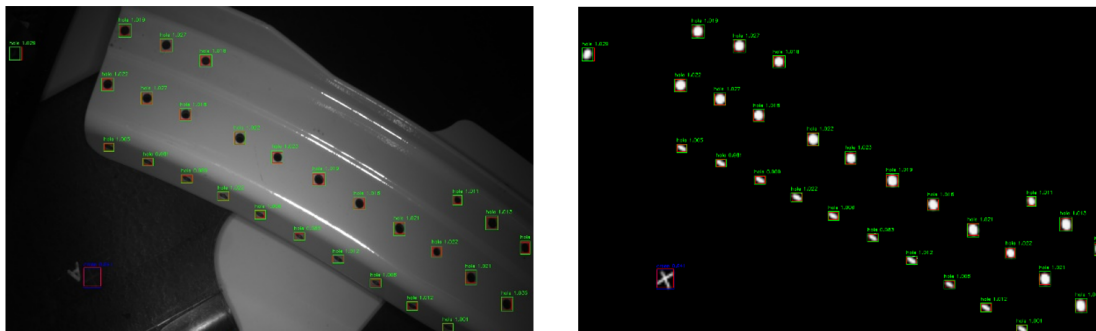
Bi metodoekin lortutako detekzioen kalitatea argi ikusten da 4.2 irudian.



**4.2 Irudia:** *Faster R-CNN* metodoarekin egindako zuloen eta gurutzeen detekzioa goran eta *HRNet*-arekin egindakoak beheran. BBox gorriak *ground truth*-a irudikatzen dute, berdeak metodoek detektatutako zuloak eta urdinak berriz, metodoek detektatutako gurutzeak. BBox gaineko zifrak BBox detekzioaren konfiantza maila adierazten du.

*HRNet* ereduaren kasuan, lortutako detekzioen segmentazio mapak 4.3 irudian ikusi daitezke. Bertan, segmentazio mapak markatzaileak oso ondo lokalizatzen dituztela ikusi daiteke, gurutzeen eta zuloen formak mantenduz.

<sup>2</sup>*HRNet* ereduaren inferentzia denbora 95ms da, ondoren BBox-en bilaketa beste 45ms direlarik.



**4.3 Irudia:** *HRNet* ereduarekin egindako zuloen eta gurutzeen detekzioa eta BBox-ak lortzeko ereduak itzulitako segmentazio mapa.

#### Erabakiak

Datu gehikuntza teknika desberdinei dagokienez, argi geratu da, arazo honetan biraketa eta iraulketak hobeto entrenatzen lagundu dutela. Ausazko zaratak ez ditu emaitza onak eman eta horregatik, baztertu egin da.

Horrez gain, emaitzak begiratzuz ikusi daiteke, metodo hauekin kamera polarimetrikoaren erabilerak eta bere fisika errespetatuz eginiko datu gehikuntza teknikak ez dutela gehiegi lagundu detekzioen emaitzak hobetzen. Metodoak irudiak tratatzeko forma edota polarizazioaren informazio gehigarririk gabe markatzaileak erraz detektatzeko ahalmena horren zergatia izan daitekeela ondorioztatu da.

Hori alde batera utzita, sarrera irudi guztiekin emaitza onak lortu dira. Horregatik, kamerak polarizazio angelu desberdinen intentsitateak itzultzen dituenaz, irudi eratorrien kalkuluan konputazio denbora ez galtzeko, balio horiek zuzenean sarrera irudi moduan erabiltzea erabaki da.

Metodoei dagokienez, argi geratu da, *Faster R-CNN* metodoak bere BBox detekzioen zehaztasuna hobea izan dela (4.1 taulan ikusi daiteke *IoU* altuagoko *mAP* emaitza hobeak direla *Faster R-CNN* metodoarekin lortutakoak). Dena dela, *HRNet* ereduak markatzaileak zertxobait hobeto detektatzen ditu (4.4 taulan ikusi daiteke, doitasun eta estaldura hobeak lortzen dituela). Bi metodoen arteko desberdintasuna ez da bat aukeratzeko modukoa izan eta azkenean, bi metodoen artean, *HRNet* aukeratu da, zehaztapen gehiagoko hurrengo pausuko detekzioan eredu honekin lortutako segmentazio mapek eman dezaketen laguntzagatik.

### 4.2.2 Detekzio zehatzak

Irudi desberdinen gainean prozedurak aplikatzean lortutako neurketen emaitzak 4.10 taulan ikusi daitezke. Gurutzeen kasuan badirudi polarizazioaren erabilerak gurutzeak desberdintzen laguntzen duela, emaitza onenak irudiaren polarizazio mailarekin (*DoP*) lortu direlako. Zuloetan berriz, angelu desberdinen intentsitate balioekin eta polarizazio mailarekin lortutako emaitzak dira onenak. Hala ere, ikusi daitezke, markatzaile guztien irudien ia erdiak baztertzen direla.

Irudia	$I_0$	$I_{45}$	$I_{90}$	$I_{135}$	$S_0$	$S_1$	$S_2$	AoP	DoP
<b>Gurutzeak</b>	0.278	0.284	0.261	0.291	0.265	0.234	0.227	0.036	<b>0.403</b>
<b>Zuloak</b>	<b>0.562</b>	0.546	0.547	0.55	0.538	0.256	0.24	0.052	0.536

**4.10 Taula:** Irudi desberdinen gainean, markatzaileen zentroa zehatz detektatzeko prozedurak aplikatzean detektatu ahal izan diren eta baztertu ez diren irudi ratioa.

Markatzaile gutxiago baztertzeko intentzioarekin irudi guztiak erabiliz lortutako emaitza guztiak batuz zein detektatzeko ratioa dagoen aztertu da. Emaidzak 4.11 taulan ikusi daitezkeen moduan, irudi desberdinekin zulo ia guztien zentroa detektatu ahal izan da. Gurutzeekin irudi gehiago baztertu dira eta soilik %66-a detektatu ahal izan da. Emaidza hauek eta irudi bakoitzarekin lortutakoak aztertuz esan daitezke, irudi sarrera desberdinetan markatzaile desberdinak argiago desberdintzen direla eta detekzio maila egoki bat lortzeko sarrera irudi desberdinak erabili beharko lirakeela.

	<b>Irudi guztiekin lortutako detektatze ratioa</b>
<b>Gurutzeak</b>	0.662
<b>Zuloak</b>	0.999

**4.11 Taula:** Sarrerako irudi desberdin guztiekin lortutako emaitzak batuz, markatzaile zentroaren detekzioaren eta beraz, baztertu ez diren ratioa.

### 4.2.3 Integrazioa

*HRNet* ereduak *Jetson Nano* mikroordenagailurekin eta ereduak entrenatutako AWS-ko GPU-arekin inferituz lortutako denborak 4.12 taulan agertzen dira. Argi eduki behar da, denbora horietan irudi original bakar batetik ereduarekin segmentazio mapak lortzea soilik sartzen dela.

Inferentzia denbora (RAM + SWAP)	GPU (AWS)	Jetson Nano
<b>Pytorch</b>	FP32	2,1s (3.3GB + 0.3GB)
	FP16	3,6s (3.2GB + 0.2GB)
<b>TensorRT</b>	FP32	1,4s (2.4GB + 0.2GB)
	FP16	<b>0,87s (2.2GB + 0.2GB)</b>

**4.12 Taula:** *HRNet* ereduaren AWS-ko GPU-arekin eta *Jetson Nano* mikroordenagailuarekin inferentzia lortutako denbora eta memoria erabilera.

Bai *Pytorch* eta baita *TensorRT* liburutegiak erabilia, FP16 moduan exekutatuak ez du emaitzen zehaztasuna jaitzi, FP32 edo FP16 moduan exekutatuak mAP balio ia berdinak lortu direlarik. Emaitzak ikusita esan daiteke *TensorRT* liburutegiak ereduaren exekuzioa optimizatzen laguntzen duela, inferentzia denbora gutxituz eta memoria gutxiago erabiliz. Kasu honetan ere ikusten da, FP16 moduan exekutatuta inferentzia denboran hobekuntza bat dagoela, emaitza ia berdinak izanda. Hala ere, argi geratzen da, *Jetson Nano*-aren GPU-a, zerbitzarietako GPU-etatik oso urrun dagoela.

Horrez gain, segmentazio mapetatik BBox-ak lortzeko prozesuak 45-50ms iraun du. Azkenik, detekzio zehatzean oro har, 50ms inguruko denbora behar du zulo markatzaile bakoitzaren zentroa bilatzeko. Hala ere, gurutzeen prozesuan 1-2s behar izan ditu zentroa bilatzeko. Azken prozesu hauek GPU-a erabiltzen ez dutenez, ez da desberdintasun handirik egon *Jetson Nano* eta AWS-ko exekuzio denboretan.

Neurtutako denboretan argi ikusten da *Jetson Nano* bezalako mikroordenagailuekin lan egiteko, proposatutako sistema gehiago optimizatu beharko litzatekeela.



## 5. KAPITULUA

---

### Ondorioak

---

Proiektu honetan, industriarako arazo zehatz batean, adimen artifizialaren zati den ikusmen artifiziala aplikatzeko jarraitu beharreko prozesu osoa burutu da. Horretarako lehenengo, literatura jarraituz kamera polarimetrikoen onurak aztertu ondoren, proiektu honetan erabiltzea erabaki da. Kamera mota horrekin helburuko markatzaileak dituzten piezen irudiak lortu dira eta hauek anotatzeko metodo egokienak aztertu dira, datu multzo bat sortuz. Ondoren, datu gehikuntza teknika desberdinak erabiliz, *Faster R-CNN* eta *HR-Net* ereduekin detekzioarako bi paradigmekin, irudietatik markatzaileen BBox-ak detektatu dira. Azkenik, metodo horien detekzioetatik, markatzaileen zentron detekzio zehatzak egiten saiatu da, ikusmen artifizialaren teknika klasikoagoak erabiliz.

Datu gehikuntza teknikengatik irudiak biratzean, ohiko AABB-ak anotatu beharrean, proiektu honetan burututako anotazioek (OBB-aren ardatzen 4 puntuak anotatzeak) entrenamenduan erabilitako AABB-en zehaztasuna handiagoa lortzen dutela erakutsi da.

Aztertu eta proposatutako polarizazioa errespetatzen duten datu gehikuntza teknika desberdinen funtzionamendu egokia ere frogatu da. Mundu errealean kamera biratuz eta proposatutako metodoarekin irudiak artifizialki biratuz, antzeko balioak lortzen direla erakutsi da. Horrez gain, RGB irudiei aplikatzen zaien biraketa eta iraulketa arruntak ez dutela polarizazioaren fisika errespetatzen frogatu da. Entrenamendu garaian, proiektu honetako datu multzoarekin biraketa eta iraulketa teknikek entrenamendua hobetzen lagundu dutela ikusi da. Ez da berdina gertatu zarata teknika aplikatzean, orokorrean emaitzak okertu dituelarik.

Proiektuan erabilitako objektuak detektatzeko metodo desberdinen emaitzek erakutsi du-

te, *Faster R-CNN* bezalako ohiko detekzio metodoak arazo honetarako markatzaileak detektatzeko erabili daitezkeela. Horrez gain, segmentazioa *HRNet* ereduarekin burutzeak eta ondoren, segmentaziotik BBox-ak lortzeak ere funtzionatu duela erakutsi da. Bi kasuetan identifikazio ratioa oso ona izan da, baina *HRNet* eredia izan da aukeratua, BBox-en informazioaz gain, prozesu osoan zehar sortutako segmentazio mapek emandako informazio gehigarriagatik. Maila honetako gaur egungo neurona-sareen entrenamenduek behar duten kalkulu potentzia eta denbora handia ere argi geratu da, hauen entrenamendurako GPU azpiegitura bat izatea edo bestela, *Cloud computing* moduko *AWS* edo *Google Colab*-en erabilera ia beharrezkoa delarik.

Metodoen detekzioetatik markatzaileen zentroak modu zehatzean lortzeko prozesuaren konplexutasuna argi geratu da. Markatzaileen tamaina txikiagatik irudi zatien kalitate txarrak eragin du kasu askotan zentroa ezin detektatu ahal izatea eta markatzaileen irudi horiek baztertu behar izatea. Hala ere, kalitate oneko irudietan mota bakoitzeko markatzaileen zentroa identifikatzeko prozesu bat definitu da, nahiko zehatza dirudien arren, bere detekzioen zehaztasuna oraindik neurtzeke dagoelarik.

Markatzaile naturalen detekzioan kamera polarimetrikoaren onurak eta bere eragina ere aztertu da. Proposatutako polarizazioa errespetatzen duten eta errespetatzen ez duten datu gehikuntza tekniken emaitzen diferentzia txikiak erakutsi du, metodoek polarizazioaren fisika ez dutela gehiegi baliatzen emaitzak hobetzeko. Ondorio hau baieztatu du, sarrera moduan polarizazio angelu desberdinen intentsitateen irudiekin eta soilik polarizazio angelu bakarreko intentsitateekin lortutako emaitzen berdintasunak. Honen arrazoia argi ez dagoen arren, metodoentzat detekzio errazegia izatea eta beraz, emaitza onak lortzeko polarizazioak emandako informazio gehigarria erabili behar ez izatea edota, aukeratutako metodoak horrelako sarrera irudietarako prestatuak ez egotea izan daitezkeela pentsatzen da. Detekzioetatik markatzaileen zentroaren bilaketan berriz, polarizazioa baliatzen duten AoP moduko irudiekin, angelu bakarreko polarizazio intentsitateen irudiekin baina emaitza hobeak lortzen dira gurutzeen zentroen detekzioan. Beraz, kasu horietan kamera polarimetrikoa erabiltzeak emaitzak hobetzen lagundu du.

Azkenik, integrazio fasean ikusi da *Jetson Nano* bezalako mikroordenagailu bat erabiliz, *HRNet* ereduak irudi batetik segmentazio mapak sortzeko gutxienez 0,87 segundo behar dituela. Inferentzia abiadura handiagoa beharko balitzake, eredu txikiagoak edota mikroordenagailu ahaltzuagoak beharko liriteke. Mapetatik BBox-ak lortzeko prozesua berriz, 45-50 ms-tan exekutatu da. Markatzaileen zentroari dagokien prozeduretan, zuloen kasuan 50 ms inguru behar izaten ditu markatzaile bakoitzaren zentroa kalkulatzeko eta gurutzeen kasuan berriz, 1-2s behar izaten ditu. Denbora horiek ikusita, gurutzeen

zentroen detekzioa gehiago optimizatu beharko litzatekeela argi geratzen da.

Proiektu hau gaur egungo ikusmen artifizialeko teknika berrienak industrian aplikatzeko formaren adibide garbia da. Horrez gain, aurretik neurona-sareak erabiltzen ez ziren metrologia bezalako arlo batean, hauek nola erabili daitezkeen erakusten da, teknika klasikoak eta teknika berrienak konbinatuz. Hala ere, kontuan eduki behar da proposatutako teknika eta prozesuen zurruntasuna eta emaitzen azalgarritasuna sakonki aztertzeko proiektu hau baino haratago joan beharko dela, egoera desberdin gehiagotan beraien funtzionamendua aztertuz.

## 5.1 Etorkizuneko lana

Proiektu hau jarraituz, interesgarriak izan daitezkeen hurrengo pausu hauek proposatzen dira:

- **Prozeduren zehaztasuna neurtu:** emaitzen atalean azaldu bezala, detekzioen zehaztasuna neurtzeko egin beharrezko prozesua oso konplexua da. Hala ere, definitutako prozeduren emaitzak neurtzeko egin beharreko lana da. Hori egiteko, CMM edo *Laser Tracker*-arekin egin beharreko demo bat muntatu beharko litzateke. Demo horretan, CMM edo *Laser Tracker*-ak emandako posiziotik, detekzioak zein errore duen ikusi behar da. Horretarako, lehenik eta behin, metodo desberdinek detektatutako BBox-en zentroa konprobatuko litzateke eta ondoren, markatzaileen zentroen detekzioak markatzailea baztertzen ez badu, detekzio finago horren errorea kuantifikatu beharko litzateke.
- **Datu gehiago lortu:** proiektu honetan 354 irudiz osatutako datu multzo murriz bat sortu da. Hala ere, proiektuarekin aurrera jarraitu eta industrializatzeko, datu multzoa handitu beharko litzateke gutxienez 1000-2000 irudi eduki arte. Bukaerako datu multzo horretako irudiak, argi desberdinekin, posizio desberdinetan eta hondo desberdinekin izan beharko lirateke. Honek lan handia suposatuko luke, irudiak ateratzeko lekua egokitu beharko litzatekeelako eta ondoren, irudi bakoitza ahalik eta zehatzenen anotatu beharko litzatekeelako. Hala ere, datu multzoa forma horretan handituko balitz, eredu eta metodoen entrenamenduan ikasitako ezaugarriak zurrungoak izango lirateke, sistemaren errepikakortasuna egoera guztietan mantenduz. Datu gehiago edukitzeak *Vision Transformer* moduko beste metodoak frogatzea ere ahalbidetuko luke, hauetako batzuekin detekzio ratioa handitzera iritsiz.

- **Irudien kalitatea hobetu:** lan honetan detektatu beharreko markatzaileen tamaina nahiko txikia da. Detekziorako erabilitako eredu eta metodoentzat errazagoa da objektu handiagoak detektatzea, objektu txikiegiak detektatzeko arazoak izaten dituztelarik. Hau konpontzeko modu bat bereizmen handiagoko kamera bat erabiltzea da; kamerak bereizmen handiagoa badauka objektuei dagozkien pixel kopurua ere altuagoa izango da. Horrez gain, markatzaileen zentroen detekzio zehatzean, irudietako markatzaileak ez ziren askotan argi ikusten eta horregatik, asko baztertu behar izan dira. Hori konpontzeko, bereizmen handiagoko kameraz gain, gainazal guztian zehar intentsitate berdinarekin argitzen duten argi difusoak erabili daitezke. Azkenik, kamerak fokuratze hobea izateko, ezaugarri desberdineko beste lente batzuk ere probatu daitezke.
- **Ohiko kamerak erabili:** esperimentazioaren emaitzetan ikusi daitekeenez, arazo honetan egindako probetan kamera polarimetroak ez dirudi gehiegi laguntzen duenik. Hori ikusita, industriarako sortuak dauden RGB edota kamera monokromatikoak proba daitezke. RGB kamerekin koloreak polarizazioak baina gehiago lagundu dezakeen ikusiko litzateke. Hau egitearen beste arrazoietako bat kostua da; kamera polarimetroek beren ezaugarri bereziengatik oso garestiak dira. Oro har, RGB edo kamera monokromatikoak gehiago erabiltzen direnez, kalitate hobeko kamerak kostu txikiagoan aurkitu daitezke. Etorkizun batean sistema hau komertzializatuko balitz, kameraren kostua jaisteak sistemaren lehiakortasuna ere hobetuko luke.
- **Markatzaileen zentroen detekzio zehatza hobetu:** proiektu honetan zehar proposatutako BBox-etatik abiatuta markatzaileen zentroa detektatzeko prozedurek markatzaile gehiegi baztertzen dituzte. Honen arrazoia, irudiko markatzaileen zatiaren bereizmen txiki eta kalitate falta da. Irudi txar horietatik, ikusmen artifizialeko teknika klasikoekin zentroen detekzio zehatz bat lortzea oso lan konplexua da. Horregatik, prozesu hauek *Key-point* moduko detekzioak burutzen dituzten ereduengatik ordezkatzeko izan daiteke aukera on bat. Honen arazo handiena, zehaztasun hain handiko anotazioak lortzeko zailtasuna da. Hori saihesteko, CMM motako makina batekin anotazio batzuk lortzeaz gain, artifizialki sortutako markatzaileekin ere entrenatu ahalko litzateke. Markatzaile sintetiko horiei zarata gehitzeaz gain, formaren desbideratzeak ere sartu beharko litzateke, mundu errealean perspektiba desberdinetatik ateratako markatzaileen argazkiak simulatuz. Modu honetan, horretarako erabiltzen den neurona-sare bat entrenatuz, zein zehaztasunera iritsi ahalko litzatekeen ikustea interesgarria izango litzateke.

- **Detekzio prozesua optimizatu:** proiektu honetan zehar egindako prozedurak optimizatu daitezke. Esperimentazioko integrazio fasean lortutako denboretan argi geratzen da, inferentzia abiadura egoki bat izateko definitutako prozedurak gehiago optimizatu behar direla. Horretarako, erabilitako metodo eta ereduak ezabatuz (*prunning*) edota metodo arinagoak erabilia emaitza onargarriak lortzen diren ala ez ikusi beharko litzateke. Horrez gain, segmentazio mapetatik BBox-ak lortzeko eta detekzio zehatzeko gurutzeen zentroak detektatzeko prozedurak ere optimizatu behar dira. Etorkizunean integrazioak edota abiadura hazkunde beharrak eraginda, *Python* lengoaiatik *C++*-era igaro ahalko litzateke, prozesuen eraginkortasuna eta beraien gaineko kontrol gehiago edukiz.



# **Eranskinak**





# A. ERANSKINA

---

## Proiektuaren kudeaketa

---

### A.1 Proiektuaren deskribapena eta helburuak

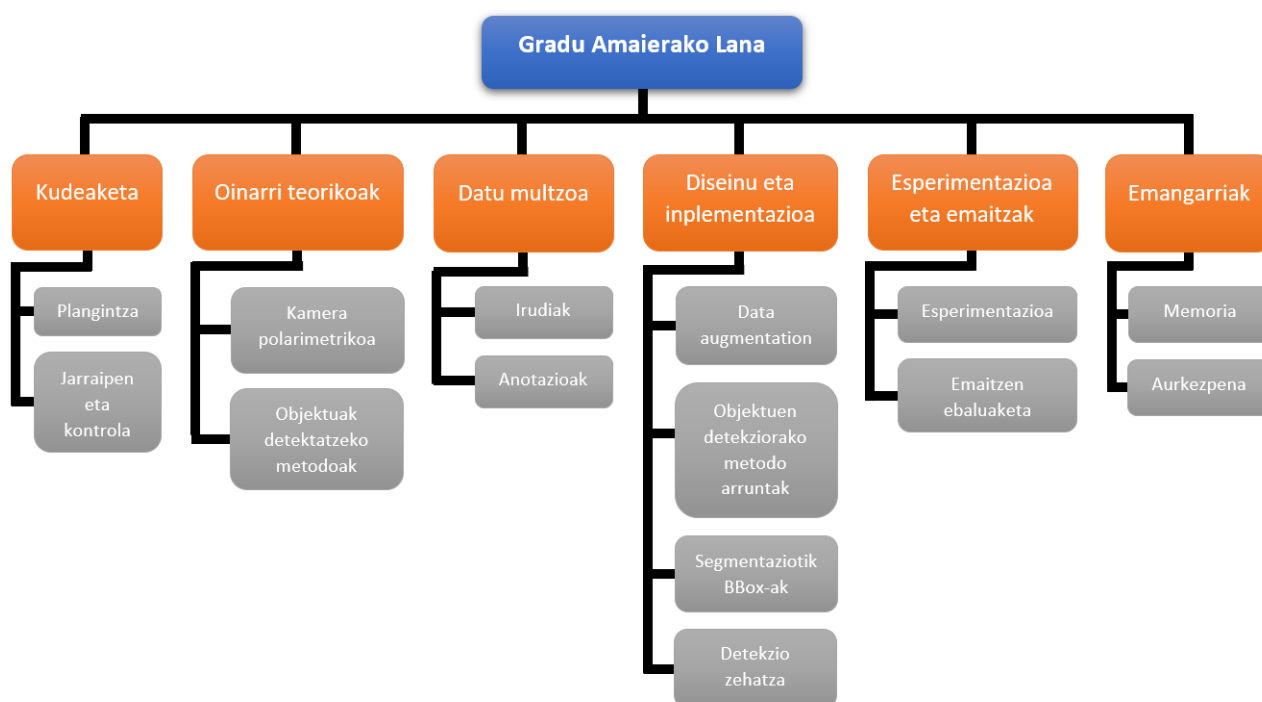
Proiektu hau hegazkinen fuselajeen zuakerren muntaketan laguntzeko ikusmen artifizial sistema bat garatzean datza. Sistema horren helburua zuaker horietan dauden markatzaile naturalen zentroa ahalik eta zehatzenen detektatzea da. Detekzio honetarako kamera polarimetroaren irudiak, objektuen detekziorako azken metodoak eta ikusmen artifizialeko teknika tradizionalak erabil behar dira. Proiektuak honako azpi-helburuak dauzka:

- Datu multzoa sortu.
- Literaturan kamera polarimetroko eta objektuen detekziorako metodoak aztertu.
- Sortutako datu multzoarekin, arazo honetarako objektuen detekziorako metodo batzuk aukeratu eta bere funtzionamendua frogatu.
- BBox detekzioetatik abiatuta, markatzaileen zentroa ahalik eta zehatzen bilatzen duten eta ikusmen tradizionaleko teknikak erabiltzen dituzten prozedurak sortu.

## A.2 Proiektuaren plangintza

### A.2.1 LDE diagrama

Proiektuaren lana Lanaren Deskonposaketa Egituraren (LDE) diagramaren bidez banatu da (A.1 diagrama).



**A.1 Irudia:** Proiektuaren lanaren LDE diagrama.

### A.2.2 Lan-paketeak

#### Kudeaketa

- **Plangintza:** atal honetan proiektuaren plangintza zehaztuko da. Horretarako, proiektuaren helburuak definitu, egin beharreko atazak sortu eta emangarriak, mugarriak eta lan-metodologia definituko dira. Honetaz gain, proiektuan eragin dezaketen arrisku eta aurrekarien analisia burutuko da.
- **Jarraipen eta kontrola:** proiektua aurrera joan ahala, planifikazioa egoki jarraitzen

ari dela egiaztatuko da. Denbora igaro ahala proiektuaren helburuak betetzen joaten direla ikusiko da. Horretarako, jarraipen bilerak erabiliko dira.

#### Oinarri teorikoak

- **Kamera polarimetrikoa:** literaturan kamera polarimetrikoa non eta zertarako erabilia izan den ikusi beharko da. Kamera berezi honen funtzionamendua eta ezaugarriak jakin beharko dira. Ondoren, ezaugarri horiek, proiektu honen arazoan nola lagundu dezaketen aztertuko da.
- **Objektuak detektatzeko metodoak:** gaur egun ikusmen artifizialean objektuak detektatzeko zein metodo eta eredu erabiltzen diren aztertuko da. Metodo eta eredu horien funtzionamendua ulertuko da eta proiektuaren markatzaileen ezaugarriak ikusirik, erabiltzeko metodo aproposenak aukeratuko dira.

#### Datu multzoa

- **Irudiak:** proiektu honetan erabili beharreko irudiak lortu beharko dira. Horretarako, kamera polarimetrikoa, argi-fokua eta lentea prestatu beharko dira, proiektuan erabiliko diren pieza desberdinen argazkiak ateratzeko. Horrez gain, irudietako piezen banaketa definituko da.
- **Anotazioak:** lortutako irudiak zein programarekin eta nola anotatu definituko da. Horretarako, kontuan eduki beharko da, anotazioak ahalik eta forma eraginkorrenean egin beharko direla eta datu multzo murriztagatik, ondoren datu gehikuntza teknikak aplikatu beharko zaizkiela irudi eta anotazioei.

#### Diseinu eta inplementazioa

- **Datu gehikuntza:** proiektuan sortuko den datu multzo murriztagatik datu gehikuntza zein teknikak eta nola aplikatu definitu eta inplementatu beharko da. Horretarako, kontuan eduki beharko da, teknika horiek kamera polarimetroarekin lortutako irudien ezaugarriak errespetatu beharko dituztela.
- **Objektuen detekziorako metodo arruntak:** oinarri teorikoak aztertuz aukeratutako objektuak detektatzeko metodo arrunten inplementazioa bilatu eta hauek en-

trenatzeko eta probatzeko implementazioak sortu beharko dira. Horrez gain, metodo hauen hiperparametroen balio onenak aukeratu beharko dira.

- **Segmentaziotik BBox-ak:** objektuak detektatzeko metodo arruntez gain, irudien segmentaziotik BBox-ak lortzeko prozedura definitu eta inplementatu beharko da. Horrez gain, burututako anotazioetatik segmentaziorako, segmentazio mapak nola sortu zehaztu beharko da.
- **Detekzio zehatza:** metodoen BBox detekzioetatik, ikusmen artifizialaren teknika tradizionalak erabiliz, hauen zentroak ahalik eta zehatzen detektatzeko prozedurak definitu eta inplementatu beharko dira.

#### Esperimentazioa eta emaitzak

- **Esperimentazioa:** proiektuan inplementatu eta erabilitako metodoak sortutako datu multzoarekin frogatzeko, esperimentazioa diseinatu eta burutu beharko da. Bertan, zein proba eta zertarako egingo diren definituko da.
- **Emaitzen ebaluaketa:** esperimentazio burutu ondoren, emaitzak ikusita zein ondorio atera daitezkeen aztertuko da. Horretarako, prozedura desberdinetan hartutako erabakiak onak izan diren eta prozedura desberdinek exekuzioan daukaten aukera desberdinetatik onenak zein diren azalduko da.

#### Emangarriak

- **Memoria:** proiektu osoaren lana deskribatzen duen dokumentua idatzi beharko da. Dokumentu honek burututako ikerketa eta lan guztiaren laburpena eta proposatutako prozeduren deskribapen zehatzak izan beharko ditu.
- **Aurkezpena:** proiektu honen defentsarako egin beharreko aurkezpenaren prestakuntza burutu beharko da. Aurkezpenaren diapositibekin batera, zein gaietan zentratu eta aurkezpenaren ordena zehaztu beharko dira. Aurkezpenean proiektuan egindako lana zein den argi geratu behar da, ahalik modu antolatuenaz azalduz.

### A.2.3 Mugarriak

Unibertsitateak ezarritako memoriaren entrega eta proiektuaren defentsaren mugarriez gain, proiektuaren esperimentazioaren amaierari ere data muga bat jarri zaio, memoria

eta aurkezpena forma egokian burutzeko denbora izateko. Proiektuaren mugariak [A.1](#) taula ikusi daitezke.

Data	Mugarria
2022/05/26	Proiektuko esperimentazioaren amaiera
2022/06/26	Memoriaren entrega
2022/07/04 - 2022/07/13	Proiektuaren defentsa

**A.1 Taula:** Proiektuaren mugariak.

#### A.2.4 Gantt diagrama

Proiektu osoan zehar lan pakete desberdinak burutu diren denbora tarteak erakusten dituen *Gantt* diagrama [A.2](#) irudian ikusi daiteke. Bertan lan pakete bakoitza noiz hasi eta noiz bukatu den ikusi daiteke.

Lan-paketea	Hasiera	Bukaera	2022							
			Urtarrila	Otsaila	Martxoa	Apirila	Maiatza	Ekaina	Uztaila	
Kudeaketa	Plangintza	2022/01/10	2022/01/20							
	Jarraipen eta kontrola	2022/01/11	2022/07/13							
Oinarri teorikoak	Kamera polarimetrikoa	2022/01/11	2022/01/25							
	Objektuak detektatzeko metodoak	2022/01/12	2022/02/17							
Datu multzoa	Irudiak	2022/02/04	2022/03/23							
	Anotazioak	2022/02/05	2022/04/07							
Diseinu eta implementazioa	Data augmentation	2022/02/18	2022/02/25							
	Objektuen detekzioarako metodo arruntak	2022/02/28	2022/04/27							
	Segmentazioetik BBox-ak	2022/03/15	2022/05/03							
	Detekzio zehatza	2022/04/21	2022/05/13							
Esperimentazioa eta emaitzak	Esperimentazioa	2022/03/04	2022/05/26							
	Emaitzen ebaluaketa	2022/04/22	2022/05/30							
Emangarriak	Memoria	2022/02/13	2022/06/26							
	Aurkezpena	2022/06/20	2022/07/13							

**A.2 Irudia:** Proiektuko *Gantt* diagrama.

## A.3 Lan metodologia

Proiektu hau IDEKO Zentro Teknologikoak bultzatua izan da. Zehazki bertako Diseinu eta Doitasun taldeko ikusmen artifizial taldean burutu da. Bertan, ikasleak Imanol Herrera Asteasu ikertzaile aditua izan du tutore moduan. Euskal Herriko Unibertsitate (EHU)-ko tutore moduan IXA taldeko Gorka Azkune Galparsoro ikertzailea izan du laguntza moduan. Proiektuan zehar, ikasleak IDEKOk eskainitako azpiegiturak (AWS eta bertako kamera eta konputagailuak) baliatu ditu.

### A.3.1 Lan eremua

Proiektu hau burutzeko, ikaslea astean bi egunez Elgoibarko IDEKOn instalazioetara joan da. Beste hiru egunetan, etxetik egin du lan.

### A.3.2 Bilerak

Ikasleak IDEKoko tutorearekin astean behin baino gehiagotan egin ditu jarraipen bilerak, Elgoibar-ko instalazioetan fisikoki eginez eta etxetik *Microsoft Teams* erabiliz. EHU-ko Gorka tutorearekin berriz, bilerak urriagoak izan dira, memoriaren eta lanaren kontzeptu orokorretan laguntzeko. Gorkarekin bilerak *Google Meet* erabiliz burutu dira.

### A.3.3 Planifikatutako ordutegiak

Lan hau burutzeko ordutegi finko bat izan da. Astelehen eta asteazkenean Elgoibarko instalazioetan egunero 8,5 ordu egin dira, beste egunetan etxetik beste 4 ordu eginez. Ikasleak bere irakasgaiak bukatu ondoren, ordutegia aldatu da.

### A.3.4 Ordu estimazioa eta erabiliak

Proiektu osoan zehar lan-pakete bakoitza burutzeko denboraren estimazioa eta benetan igarotako denborak [A.2](#) taulan agertzen dira. Ikusten den moduan, estimatutako denbora baino gehiago behar izan da, lan sakona egin delako eta batez ere, datu multzoaren eraikuntzan eta esperimentazioan estimatua baina denbora gehiago behar izan delako.

<b>Lan paketea</b>	<b>Estimaturia</b>	<b>Erabilia</b>
<b>Kudeaketa</b>	<b>25</b>	<b>40</b>
Plangintza	5	10
Jarraipen eta kontrola	20	30
<b>Oinarri teorikoak</b>	<b>60</b>	<b>70</b>
Kamera polarimetrikoa	20	25
Objektuak detektatzeko metodoak	40	45
<b>Datu multzoa</b>	<b>30</b>	<b>55</b>
Irudiak	20	25
Anotazioak	10	30
<b>Diseinu eta inplementazioa</b>	<b>90</b>	<b>110</b>
Datu gehikuntza	10	20
Objektuen detekzioarako metodo arruntak	35	30
Segmentaziotik BBox-ak	5	10
Detekzio zehatza	40	50
<b>Esperimentazioa eta emaitzak</b>	<b>75</b>	<b>105</b>
Esperimentazioa	70	100
Emaitzen ebaluaketa	5	5
<b>Emangarriak</b>	<b>100</b>	<b>115</b>
Memoria	90	100
Aurkezpena	10	15
<b>Total</b>	<b>380</b>	<b>495</b>

**A.2 Taula:** Proiektuko lan-pakete bakoitzeko estimaturako denbora eta benetan behar izan den denbora.

## A.4 Arriskuak eta hartutako erabakiak

Proiektua aurrera eramatea oztopatu dezaketen hainbat arrisku daude. Horietako bakoitzerako, arriskua identifikatu ondoren, arriskuaren aurrean, bere eragina oztopatzeko erabakiak hartu dira. Hauek izan dira identifikatutako arrisku nagusiak:

### A.4.1 Entrenamendu luzeak

- **Deskribapena:** proiektu honetan bereizmen handiko irudiekin neurona-sare oso handiak entrenatuko direnez, entrenamendu bakoitzaren denbora handia izango da. Hori kontuan hartuta, ezin izango da GPU erabilpen mugak dauzkaten *Google Colab* moduko doako zerbitzuak erabili. Honek eragingo du, esperimentazioan zehar entrenamendu asko egin beharko direnez, esperimentazioa ongi burutzeak kostu handia izatea eta kasu larrietan behar diren baino entrenamendu gutxiago egitea.
- **Hartutako erabakiak:** arrisku hauen aurrean, proiektu hau IDEKO zentro teknologikoan egin denez, hauek AWS zerbitzuko instantziak erabiltzea ahalbidetu dute. Instantzia hauek GPU berri eta eraginkorrenak daukate, entrenamendu denbora ahalik eta txikiena izateko. Horrez gain, instantzia hauen orduko kostua IDEKO zentro teknologikoak babestu du, behar izan diren entrenamendu guztiak egitea ahalbidetuz.

### A.4.2 Proiektu handiegia

- **Deskribapena:** proiektu hau IDEKO zentro teknologikoan egingo denez eta bertako proiektu baten parte izango denez, gerta daiteke proiektu sakon eta handiegi batean sartzea. Honek eragingo luke, mugarriak betetzeko zailtasun handiak izatea, proiektuko lan eta memoriaren kalitatea jaitsiz.
- **Hartutako erabakiak:** arrisku hau oztopatzeko, lehen momentutik IDEKOk Imanol tutoreari mugarrien berri eman zaio eta proiektua burutu ahala, jarraipen bileretan mugarri horiek errespetatzeko, hasierako lanaren puntuak kendu edo gehitu eztabaidatu da. Horrez gain, IDEKOk independentea den IXA taldeko Gorka Azkune tutoreari ere proiektuaren nondik norakoan berri eman zaio, honek beti mugarriak errespetatzeko bideragarritasuna baieztatu duelarik.



### A.4.3 COVID-19 pandemia

- **Deskribapena:** garai hauetan beti kontuan eduki behar da, COVID-19 birusa oraindik populazioaren artean dagoela. Baliteke positiboen tasa handitzea edota birusarekin infektatzea eta ezin fisikoki IDEKOk Elgoibarko instalazioetara joatea proiektua burutzeko. Honek, proiektuan atzerapen larriak ekarri ahalko lituzke.
- **Hartutako erabakiak:** arrisku hau oztopatu ezin denez, hori gertatuko balitz, lan telematikoa egiteko prestakuntzak egin dira. Horretarako, lanaren datu guztiak hodeian gorde dira eta komunikazioan laguntzeko Microsoft Teams eta Google Meet bidezko bilerak prestatu dira. Horrez gain, esperimentazioa AWS zerbitzuan burutu denez, edozein kasutan ere proiektua etxetik ere aurrera eramatea ahalbidetu du.



---

## Bibliografia

---

- [Al-Battal et al., 2021] Al-Battal, A. F., Gong, Y., Xu, L., Morton, T. D., Du, C., Bu, Y., Lerman, I. R., Madhavan, R., and Nguyen, T. Q. (2021). A cnn segmentation-based approach to object detection and tracking in ultrasound scans with application to the vagus nerve detection. *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 3322–3327.
- [Blanchon et al., 2020] Blanchon, M., Morel, O., Meriaudeau, F., Seulin, R., and Sidibé, D. (2020). Polarimetric image augmentation.
- [Blin et al., 2019] Blin, R., Ainouz, S., Canu, S., and Meriaudeau, F. (2019). Road scenes analysis in adverse weather conditions by polarization-encoded images and adapted deep learning.
- [Cheng et al., 2019] Cheng, Z., Wu, Y., Xu, Z., Lukasiewicz, T., and Wang, W. (2019). Segmentation is all you need.
- [Dalal and Triggs, 2005] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893 vol. 1.
- [Felzenszwalb et al., 2010] Felzenszwalb, P., Girshick, R., Mcallester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32:1627–45.
- [Garcia et al., 2015] Garcia, N. M., de Erausquin, I., Edmiston, C., and Gruev, V. (2015). Surface normal reconstruction using circularly polarized light. *Opt. Express*, 23(11):14391–14406.
- [Girshick, 2015] Girshick, R. (2015). Fast r-cnn.

- [Girshick et al., 2013] Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2013). Rich feature hierarchies for accurate object detection and semantic segmentation.
- [Goldstein and Collett, 2003] Goldstein, D. H. and Collett, E. (2003). *Polarized light*. New York : Marcel Dekker.
- [He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition.
- [Hough, 1962] Hough, P. V. C. (1962). *Method and means for recognizing complex patterns*. U.S. Patent 3,069,654.
- [Huber, 1964] Huber, P. J. (1964). Robust estimation of a location parameter. *Annals of Mathematical Statistics*, 35:492–518.
- [Kalra et al., 2020] Kalra, A., Taamazyan, V., Rao, S. K., Venkataraman, K., Raskar, R., and Kadambi, A. (2020). Deep polarization cues for transparent object segmentation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8599–8608.
- [Kingma and Ba, 2014] Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *International Conference on Learning Representations*.
- [Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc.
- [Lacombe et al., 2019] Lacombe, T., Nagorny, P., Pillet, M., Favreliere, H., and Pairel, E. (2019). Polarimetric imaging for quality control in injection molding.
- [Lei et al., 2018] Lei, H.-W., Wang, B., Wu, H.-H., and Wang, A.-H. (2018). Defect detection for polymeric polarizer based on faster r-cnn. *Journal of Information Hiding and Multimedia Signal Processing*, 9:1414–1420.
- [Lin et al., 2016] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2016). Feature pyramid networks for object detection.
- [Lin et al., 2014] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In Fleet, D., Pajdla, T., Schiele, B., and Tuytelaars, T., editors, *Computer Vision – ECCV 2014*, pages 740–755, Cham. Springer International Publishing.

- [Liu et al., 2021] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows.
- [Moreno-Barea et al., 2018] Moreno-Barea, F. J., Strazzera, F., Jerez, J. M., Urda, D., and Franco, L. (2018). Forward noise adjustment scheme for data augmentation. In *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 728–734.
- [Otsu, 1979] Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66.
- [Pierangelo et al., 2013] Pierangelo, A., Nazac, A., Benali, A., Validire, P., Cohen, H., Novikova, T., Ibrahim, B. H., Manhas, S., Fallet, C., Antonelli, M.-R., and Martino, A.-D. (2013). Polarimetric imaging of uterine cervix: a case study. *Opt. Express*, 21(12):14120–14130.
- [Qi and Elson, 2017] Qi, J. and Elson, D. S. (2017). Mueller polarimetric imaging for surgical and diagnostic applications: a review. *Journal of Biophotonics*, 10(8):950–982.
- [Ren et al., 2015] Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks.
- [Saito et al., 2001] Saito, M., Sato, Y., Ikeuchi, K., and Kashiwagi, H. (2001). Measurement of surface orientations of transparent objects using polarization in highlight. *Systems and Computers in Japan*, 32(5):64–71.
- [Uijlings et al., 2013] Uijlings, J., Sande, K., Gevers, T., and Smeulders, A. (2013). Selective search for object recognition. *International Journal of Computer Vision*, 104:154–171.
- [Viola and Jones, 2001] Viola, P. and Jones, M. (2001). Robust real-time object detection. volume 57.
- [Wang et al., 2019] Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., and Xiao, B. (2019). Deep high-resolution representation learning for visual recognition.
- [Yan et al., 2022] Yan, H., Zhang, C., and Wu, M. (2022). Lawin transformer: Improving semantic segmentation transformer with multi-scale representations via large window attention.

[Zhang and Suen, 1984] Zhang, T. Y. and Suen, C. Y. (1984). A fast parallel algorithm for thinning digital patterns. *Commun. ACM*, 27:236–239.

[Zou et al., 2019] Zou, Z., Shi, Z., Guo, Y., and Ye, J. (2019). Object detection in 20 years: A survey.