

**VENTSPILS AUGSTSKOLA
INFORMĀCIJAS TEHNOLOĢIJU FAKULTĀTE**

KURSA PROJEKTS

**Pētījums par radiofrekvences traucējumu mazināšanas metodēm, darbojoties ar
LOFAR datiem**

Autors:

Ventspils Augstskolas
Informācijas tehnoloģiju fakultātes
maģistra studiju programmas
„Datorzinātnes”
2 kursa students
Emīls Riekstiņš
Matr.nr. 21080006

Zinātniskais vadītājs:

lektore un pētniece, Mg.sc.comp, Karina Šķirmante

Ventspils
2023

SATURS

1. IEVADS.....	3
2. LOFAR DATI.....	4
2.1. LOFAR datu iegūšana.....	4
2.2. LOFAR datu apstrāde	6
3. REZULTĀTU PRECIZITĀTES NOTEIKŠANAS METODES	9
3.1. RMSE.....	9
3.2. PSNR.....	10
4. RFI NOTEIKŠANAS METOŽU PRAKTISKAIS PIELIETOJUMS.....	11
4.1. Parastā sliekšņošana	11
4.2. Mediānas filtrs.....	12
4.3. Vienotais filtrs	15
4.3. Gausa filtrs	16
4.5. Minimuma filtrs	17
4.6. SumThreshold metode	18
4.6.1. SumThershold darbības princips	19
4.6.2. SumThershold metodes praktiskais pielietojums	20
4.7. Kalibrēšana ar off_target.....	23
5. SECINĀJUMI UN PRIEKŠLIKUMI.....	32
IZMANTOTĀS LITERATŪRAS UN AVOTU SARAKSTS.....	35
A .sh faila piemērs LOFAR novērojumu veikšanai	37
B Izmantotās PSNR un RMSE precizitāšu noteikšanas un SumThreshold metodes funkcijas	37

1. IEVADS

Zemfrekvences masīvs jeb LOFAR (*angl. Low Frequency Array*) ir lielākais radioteleskops, kas uztver astronomiskos objektus zemās frekvencēs. LOFAR ir daudzfunkcionāls sensoru tīkls ar savienotu tīkla infrastruktūru, ar kura palīdzību iespējams apstrādāt lielus datu daudzumus. Desmit gadu laikā LOFAR teleskopi ir tikuši izvietoti deviņās Eiropas valstīs, tai skaitā arī Latvijā (LOFAR Latvia LV614 stacija Irbenē). LOFAR var tik izmantota gan viena teleskopa režīmā, gan arī LOFAR teleskopa tīkla režīmā. Piemēram, izmantojot vairākus LOFAR teleskopus, lielais attālumu diapazons starp teleskopiem nodrošina detalizētu debesu attēlu ieguvu konkrētās zemajās frekvencēs. [1]

LOFAR izmanto fāzētu masīvu dizainu (*angl. phased-array*) un stacijas sastāv no daudzām mazākām antenām, kas signālus uztver no visiem virzieniem. Tas nosedz iepriekš lielākoties neizpētītu zemfrekvenču diapazonu 10-240 MHz. Pateicoties tā blīvajam kodolu masīvam, LOFAR nodrošina vāju signālu novērojumu datu iegūšanu. [2]

Diemžēl tieši jūtīgo antenu dēļ rodas problēmas ar tīru derīgā signālu uztveršanu no astronomiskajiem objektiem. Tāpat kā citi radioteleskopi, arī LOFAR papildus derīgajiem signāliem uztver radiofrekvences traucējumus jeb RFI (*angl. Radio Frequency Interference*), kas veic datu piesārņojumu. RFI rodas no dažādiem avotiem, kuri var būt gan cilvēku veidoti, kā frekvences modulācijas (FM) radio transmisijas (87.5 – 108 MHz diapazonā) un televīzijas iekārtas (40 – 80 MHz, 160 – 230 MHz), gan dabas parādības, kā zibens, saules starojums un kosmiskais troksnis ārpus Zemes atmosfēras. [3], [4]

Šim iemeslam, lai no radioastronomiskajiem datiem būtu iespējams iegūt tīru derīgo signālu, tiek pielietotas dažādas RFI detektēšanas un noņemšanas metodes. Iegūt tīrus derīgus datus ir ļoti nozīmīgi, jo nekvalitatīvu datu gadījumā iegūtais gala rezultāts arī ir nekorekts, taču arī attīrīšanas metodes ir jāizvēlas efektīvas, lai arī sākotnējais signāls netiktu kropļots.

RFI mazināšanas metodes var būt dažādas sliekšņošanas un filtrēšanas pieejas, un pētījumā sīkāk tika apskatītas un praktiski realizētas mediānas, Gausa, minimuma un vienotais filtrs, izmantota *SumThreshold* sliekšņošanas metode un izvirzīta jauna pieeja RFI mazināšanai LOFAR attēlos ar kalibrēšanu, kā arī analizētas šo metožu pozitīvās un negatīvās puses.

2. LOFAR DATI

Ventspils Starptautiskais Radioastronomijas Centrs (VSRC) darbinieki sākuši izmantot LOFAR teleskopu tikai nesen, organizējot savus novērojumus un pētot dažādus kosmosa objektus, to skaitā Sauli, Cassiopeia A zvaigznes supernovas paliekas (turpmāk CASA), Jupiteru, kā arī kosmosa daļas, kuras pēc iespējas mazāk satur kosmiskos objektus un dažāda veida starojumus, ko izmantot kalibrēšanas un savstarpējo rezultātu salīdzinājumu veikšanai (turpmāk *off_target*). Vienā novērojumā vienlaicīgi nereti tiek novēroti vairāki debess ķermeņi un kalibrēšanas apgabals (*off_target*), jo LOFAR antenu masīvs tiek sadalīts vairākās daļās, katru daļu vērst uz konkrētām objekta koordinātām.

LOFAR dati tiek iegūti divos datu formātos. Viens ir tā sauktais "*beam formed statistics*" jeb BST, kas tiek pielietots, ja novērojuma integrācijas laiks ir viena sekunde. Otrs ir "*beam formed streams*", kuru pielieto, ja integrācijas laiks ir zem vienas sekundes. "Beam formed streams" tiek izmantots, lai pētītu tādus notikumus, kā zibens izlādes uz Jupitera, jo tā norises laiks ir mazāks par vienu sekundi. Pētījumā izmantotajos datos visu objektu novērojumi bija saglabāti BST formātā.

2.1. LOFAR datu iegūšana

LOFAR datu īpašība ir tāda, ka tos iespējams attēlot divās dimensijās jeb no tiem var izveidot attēlus. Šos attēlus parasti attēlo uz vienas ass norādot laika dimensiju, uz otras – frekvenču diapazonu. Pētījumā izmantotie dati ir iegūti no LOFAR Latvia LV614 stacijas.

Datu iegūšanai tiek izmantots *shell* skripta fails jeb *.sh*, kurā norādīti nepieciešamie parametri objektu novērošanai. Galvenie parametri ir [5]:

1. RCU mode – *receiver unit* jeb uztvērēja vienības režīms, kurš norāda, kādas frekvences tiks uztvertas (skat 1. tab.).

1. tabula. LOFAR RCU režīmi.

Režīms	Frekvenču diapazons [MHz]	Aproksimēts patiesais caurlaides joslas diapazons [MHz]	Uztvērēju joslas diapazons [MHz]
7	240-270	215-260	200-300
6	170-230	170-220	160-240
5	110-190	115-185	200-100
4	30-90	30-80	0-100
3	10-90	10-80	0-100
2	30-90	30-80	0-100
1	10-90	10-80	0-100
0	-	-	-

2. duration – objekta pētīšanas laiks sekundēs;
3. datapath – direktorija rezultātu saglabāšanai;
4. pētāmo objektu koordinātas – lai norādītu, kurā virzienā atrodas pētāmais objekts.

Piemēram:

```
CASA="6.123487680622105,1.0265153995604648,J2000"
OFF_TARGET="6.218333453664764,0.9920527098688247,J2000"
SUN="0,0,SUN"
```

5. Biti – lai norādītu, cik spektra punktus ierakstīt katrā sekundē
 - a. Ja 8 biti, tad 488 punkti
 - b. Ja 16 biti, tad 244 punkti

Tālāk tiek veikta katra objekta novērojumu veikšana, izmantojot *beamctl* staru veidošanas komandu un norādīti sekojošie parametri [5]:

1. --antennaset – antenu komponentu izvēle;
2. --rcus – norāda uztvērējus, kuri tiks izmantoti starojuma veidošanai;
3. --band – RCU filtru režīms;
4. --subbands – norāda, kuras uztvērēja apakšjoslas tiks izmantotas novērojumu veikšanai;
5. --beamlets – norāda kuri stari pieder kuram objektam. Staru skaitam jāatbilst apakšjoslu skaitam
6. --anadir – analogās augstjoslu antenu (*angl. High-band antenna* jeb *HBA*) mērījumu virziens (platums, garums, tips[ilgums]):
 - a. Platums un garums – ģeogrāfiskās koordinātas grādos
 - b. Tips – koordinātu sistēmas nosaukums
 - c. Ilgums – opcionāls parametrs tipa vietā, kas norāda stara ilgumu sekundēs

Piemēram:

```
CASA="6.123487680622105,1.0265153995604648,J2000"
```

7. --digdir – digitālās stacijas mērījumu virziens (platums, garums, tips[ilgums]):

Novērojumu veikšanas faila piemērs attēlots pielikumā A. Rezultātā iegūti norādīto objektu novērojumi, kas saglabāti divos *.dat* failos, kur katrs no tiem, respektīvi, atbilst x un y polarizācijai. Failu struktūra ir A_B_C_D.dat, kur

1. A – mērījuma veikšanas datums;
2. B – mērījuma veikšanas sākuma laiks;

3. C – datu tips;
 4. D – polarizācija,
- piemēram, 20220813_060739_bst_00X.dat un 20220813_060739_bst_00Y.dat

2.2. LOFAR datu apstrāde

LOFAR datu pirmsapstrādes kods ir iegūts no VSRC darbinieka Jāņa Šteinberga privātās *github* repozitorijas un šajā pētījumā kods tika papildināts ar izpētītajām RFI mazināšanas metodēm un analizēti to rezultāti.

Iegūtie BST datu tipa faili tiek apstrādāti un no tiem tiek iegūti dati par katru no pētītajiem objektiem (skat. 1. un 2. att.). Šajā gadījumā tie ir CASA, Saule un *off_target*.

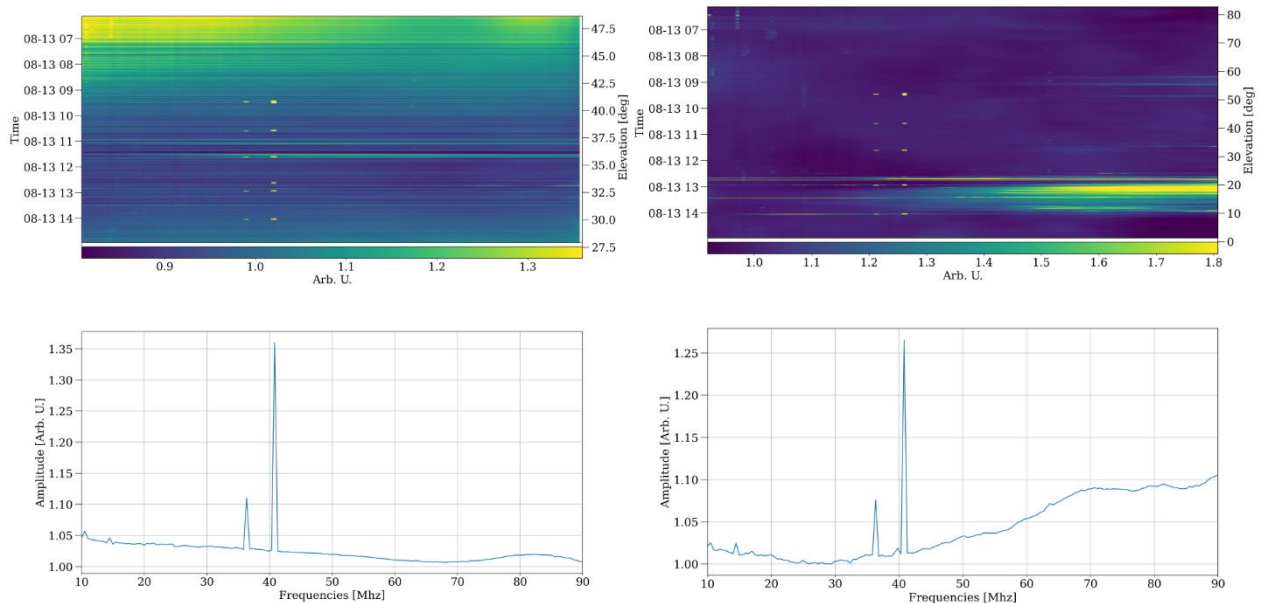
Katra objekta pirmajā (augšējā) grafikā attēlots dinamiskais spektrs, kur uz x ass attēlots frekvenču domēns, kur norādītas frekvences no 10-90 MHz un uz y ass – laika domēns, kur katra horizontālā līnija atbilst vienai sekunde. Papildus norādīts novērojumu veikšanas datums (šajā gadījumā 2022-08-13) un laiks pa stundām (no 06:00 līdz 15:00). Papildus ar krāsām ir attēlotas frekvenču intensitātes jeb amplitūda, kam nav mērvienība (*arbitrary unit* jeb *Arb. u*), kā arī pacēlums (*Elevation [deg]*) jeb cik augstu virs teleskopa objekts atrodas, taču pacēlums šī darba ietvaros tālākiem pētījumiem un datu apstrādei netiek izmantots.

Katra objekta otrajā (apakšējā) grafikā attēlotas vidējās vērtības katrā no frekvencēm norādītajā frekvenču diapazona frekvencēm. Uz x ass attēlotas frekvences un uz y – amplitūdas jeb signālu intensitātes šajās frekvencēs.

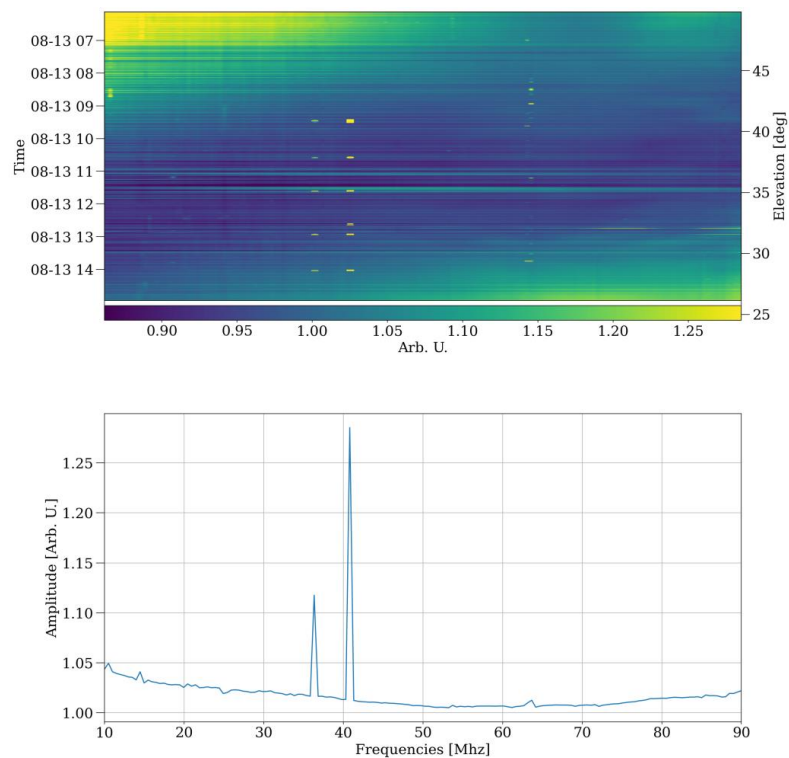
Šajos piemēra attēlos skaidri redzami RFI, kuri ierakstīti ap 36 un 41 MHz frekvencēm, kā arī mazāk izteikti traucējumi ap 11, 15 un 61 MHz. Viena no metodēm, kā noteikt vai novērotais signāls attēlā ir RFI, nevis derīgi dati, ir vizuāli salīdzināt visu novērojumu attēlus. Ja signāls pamanāms visos attēlos, tad to visticamāk var klasificēt kā traucējumu, jo LOFAR sensori pēta atšķirīgus kosmosa apgabalus dažādos debesu virzienos. Ja signāls redzams visos attēlos, tad tas visticamāk ir kāds iekšējais traucējumu veidotājs, kā pati LOFAR aparatūra, kas konkrētā brīdī sākusi funkcionēt kļūdaini, vai dažādas elektriskās ierīces ap to.

Otrs paņēmieni ir objekta novērojumu spektrus salīdzināt tieši ar *off_target* spektru. Teorētiski *off_target* attēlam nevajadzētu saturēt nekādus datus, jo sensoriem, kuri šo novērojumu ir veikuši, vajadzētu ierakstīt tikai tās kosmosa daļas, kuras nekādus debesu objektus nesatur. Šajā piemērā redzams, ka *off_target* tomēr satur kādu informāciju. Tā var

būt informācija par kādiem objektiem, kas tajā brīdī šķērsojuši LOFAR sensoru pētāmo zonu, vai arī novērojuma veikšanai nav izvēlēts labākais kosmosa apgabals.



1. att. CASA (pa kreisi) un Saules (pa labi) no BST failiem iegūtais spektrs, kas iegūts 2022-08-13 ar sākuma laiku 06:07:39 ar kopējo novērojuma laiku 31800 sekundes.



2. att. *Off_target* no BST failiem iegūtais spektrs, kas iegūts 2022-08-13 ar sākuma laiku 06:07:39 ar kopējo novērojuma laiku 31800 sekundes.

4. nodaļā tiks izmantotas dažādas metodes, lai attēlus atbrīvotu no RFI, kā arī salīdzinātas to veikspējas, izmantojot 3. nodaļā aprakstītās attēlu attrokšņošanas precizitātes noteikšanas metodes.

3. REZULTĀTU PRECIZITĀTES NOTEIKŠANAS METODES

Lai noteiktu, cik labi apstrādātie attēli ir attīrīti no RFI un vai pielietotās RFI mazināšanas metodes ir sniegušas vēlamos rezultātus, tiek pielietotas dažādas rezultātu precizitātes noteikšanas metodes. Radioastronomijā precizitāti nosaka, izmantojot RMSE (vidējo kvadrātisko kļūdu) un SNR (signāla un trokšņa savstarpējo attiecību).

3.1. RMSE

Vidējā kvadrātiskā kļūda aprēķina kļūdas iegūtajos datos, veicot to salīdzināšanu ar prognozētajiem datiem, kuri ir tīri no RFI. RMSE ir prognozēšanas kļūdu jeb atlikumu standartnovirze. Iegūtais rezultāts pats par sevi neko nenožīmē, taču veicot to salīdzināšanu starp visiem rezultātiem, kas iegūti izmantojot dažādas RFI noņemšanas metodes, visprecīzākais būs rezultāts ar mazāko RMSE vērtību. RMSE tiek aprēķināta pēc formulas [6]:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (f_i - o_i)^2} \quad (3.1.),$$

kur:

n – signāla garums

f_i – reāli iegūtais signāls

o_i – prognozētais signāls

Attēlapstrādē reāli iegūtā signāla vietā var izmantot attīrīto attēlu (ar samazinātu RFI) un prognozētā signāla vietā var izmantot oriģinālo attēlu (ar RFI), pielāgojot formulu tieši attēlu analizēšanai [9]:

$$RMSE = \sqrt{\frac{\sum_{i=1}^m \sum_{j=1}^n [M(i, j) - F(i, j)]^2}{m \cdot n}} \quad (3.2.),$$

kur:

m, n – attēla izmērs

i, j – pikseļu pozīcijas attēlā

$M(i, j)$ – attīrītais attēls (bez RFI)

$F(i, j)$ – oriģinālais attēls

3.2. PSNR

Vēl viena no pielietotajām metodēm ir SNR (*angl. signal-to-noise ratio*) jeb signāla un trokšņa savstarpējā attiecība. Tā ir attiecība starp vēlamo signālu un fona troksni, un to parasti izsaka decibelos (dB). Lai varētu izmantot SNR, tad gan troksnim, gan derīgajam signālam ir jābūt uzņemtiem ar vienādu mērvienību un jāmēra tajos pašos vai līdzvērtīgos atskaites sistēmas punktos. [7], [8]

SNR tiek realizēts pēc formulas:

$$SNR_{db} = \frac{P_{signāls}}{P_{troksnis}} \quad (3.3.),$$

kur:

$P_{signāls}$ = signāla amplitūdas vērtības

$P_{troksnis}$ = trokšņa amplitūdas vērtības

Kompleksu kalkulāciju veikšanai SNR nepieciešams pārveidot decibelu formā. Tas iespējams, izmantojot formulu:

$$SNR_{db} = 10 \log_{10} \frac{P_{signāls}}{P_{troksnis}} \quad (3.4.),$$

Attēlapstrādē SNR tiek pielāgots, lai attiecību varētu noteikt tieši uz pašiem attēliem. Izmantotā metode tiek saukta par PSNR (*angl. peak signal-to-noise ratio*) jeb maksimuma signāla un trokšņa savstarpējā attiecību. Jo lielāka PSNR vērtība, jo efektīvāka bijusi attēla attīrīšana. [9]

Tas tiek aprēķināts ar formulu:

$$PSNR = 10 \log_{10} \left(\frac{f_{max}}{RMSE} \right)^2 \quad (3.5.),$$

kur:

f_{max} = maksimālā pikseļa vērtība attīrītajā attēlā.

Lai abas no šīm metodēm korekti varētu izmantot uz LOFAR attēliem, ir jāveic gan sākotnējā, gan no RFI attīrītā attēla normalizācija. Tas tika veikts, normalizējot attēlu vērtības intervālā no 0 līdz 1.

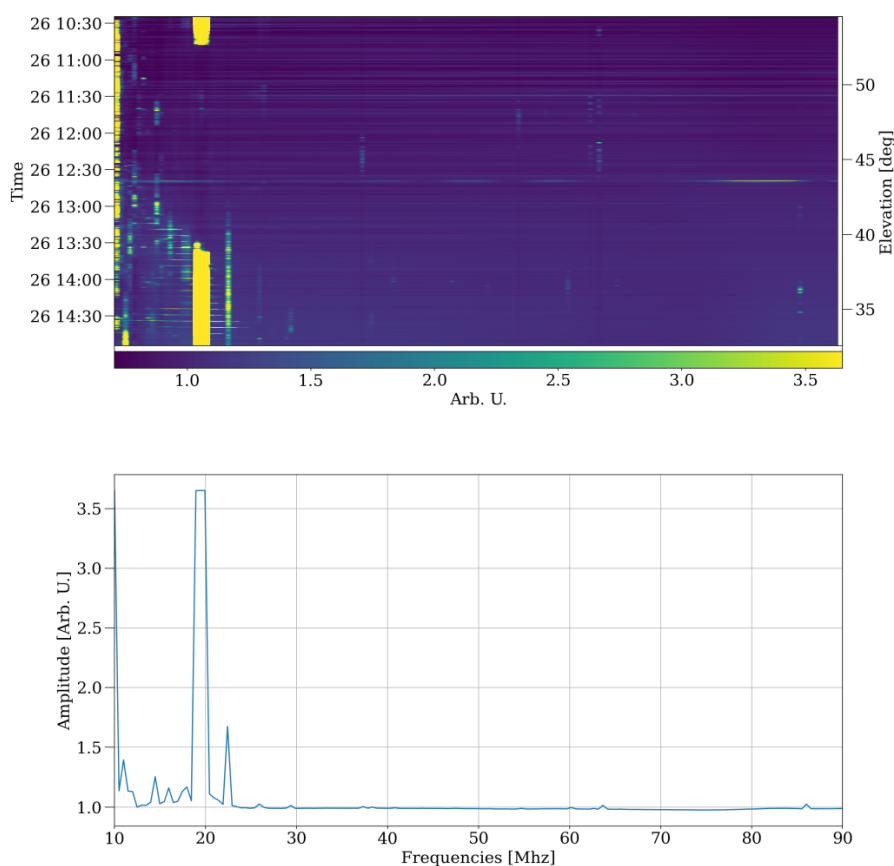
4. RFI NOTEIKŠANAS METOŽU PRAKTISKAIS PIELIETOJUMS

4.1. Parastā sliekšņošana

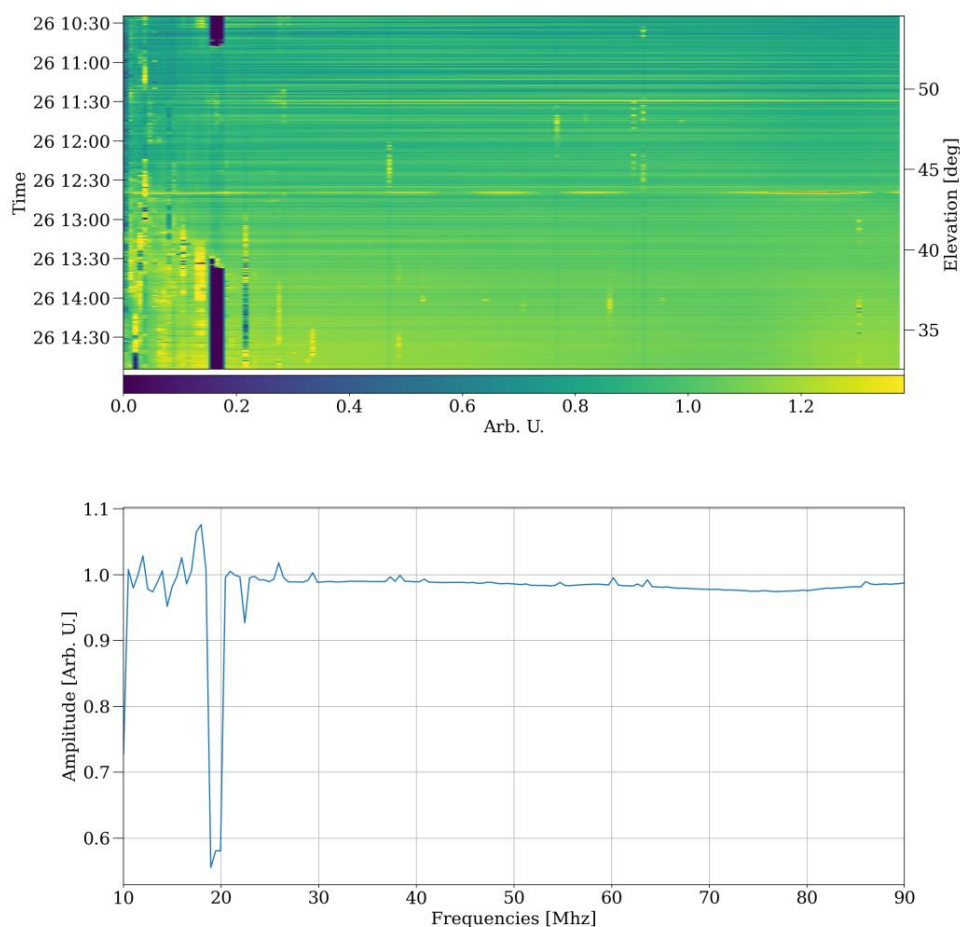
Sliekšņošanas metodes parasti darbojas uz tāda principa, ka ja vērtību amplitūdas ir virs kāda iepriekš definēta sliekšņa, tad tās tiek klasificētas kā troksnis. Izvēlēta sliekšņa vērtība ir jāizvēlas tāda, kas ir pielāgota katram attēlam individuāli. Tā var būt kāda statistiska manuāli izvēlēta vērtība vai statistiski aprēķināts skaitlis, kā vidējā vērtība vai mediāna.

Kā pirmais un vienkāršākais RFI mazināšanas paņēmiens tika pārbaudīta parastā sliekšņošana. Darbības princips ir izvēlēties kādu sliekšņa vērtību un pielietot to visam LOFAR attēlam.

Testēšanai tika izvēlēts attēls, kurā ir izteikti pamanāms RFI un tas nav sajaucams ar citu signālu (skat. 3. att.). Kā sliekšņošanas vērtība tika izvēlēta vidējā vērtība. Pēc attēla sliekšņošanas tika noņemtas visas vērtības, kuras pārsniedz šo sliekšni tika pārveidotas par nullēm (skat. 4. att.). Kā redzams metode atstāja arī visas trokšņa vērtības, kuras ir mazākas par norādīto sliekšni.



3. att. CASA (2022-10-26_10:24:33 ar novērojuma laiku 16200 sekundes) spektrs pirms
sliekšņošanas pielietošanas RFI mazināšanai.



4. att. CASA (2022-10-26_10:24:33 ar novērojuma laiku 16200 sekundes) spektrs pēc sliekšņošanas pielietošanas RFI mazināšanai.

Izmēģinot metodi uz citiem LOFAR attēliem, kuru RFI nav tik izteikts un kuri satur citus derīgos signālus ar augstām amplitūdām, šī metode nebija efektīva, jo papildus RFI tika noņemta arī derīgā informācija. Šim iemeslam nepieciešams pētīt tieši RFI raksturīgas īpašības un veikt sliekšņošanu balstoties uz tām. Šo procesu veic 4.6. apakšnodaļā aprakstītā *SumThreshold* metode.

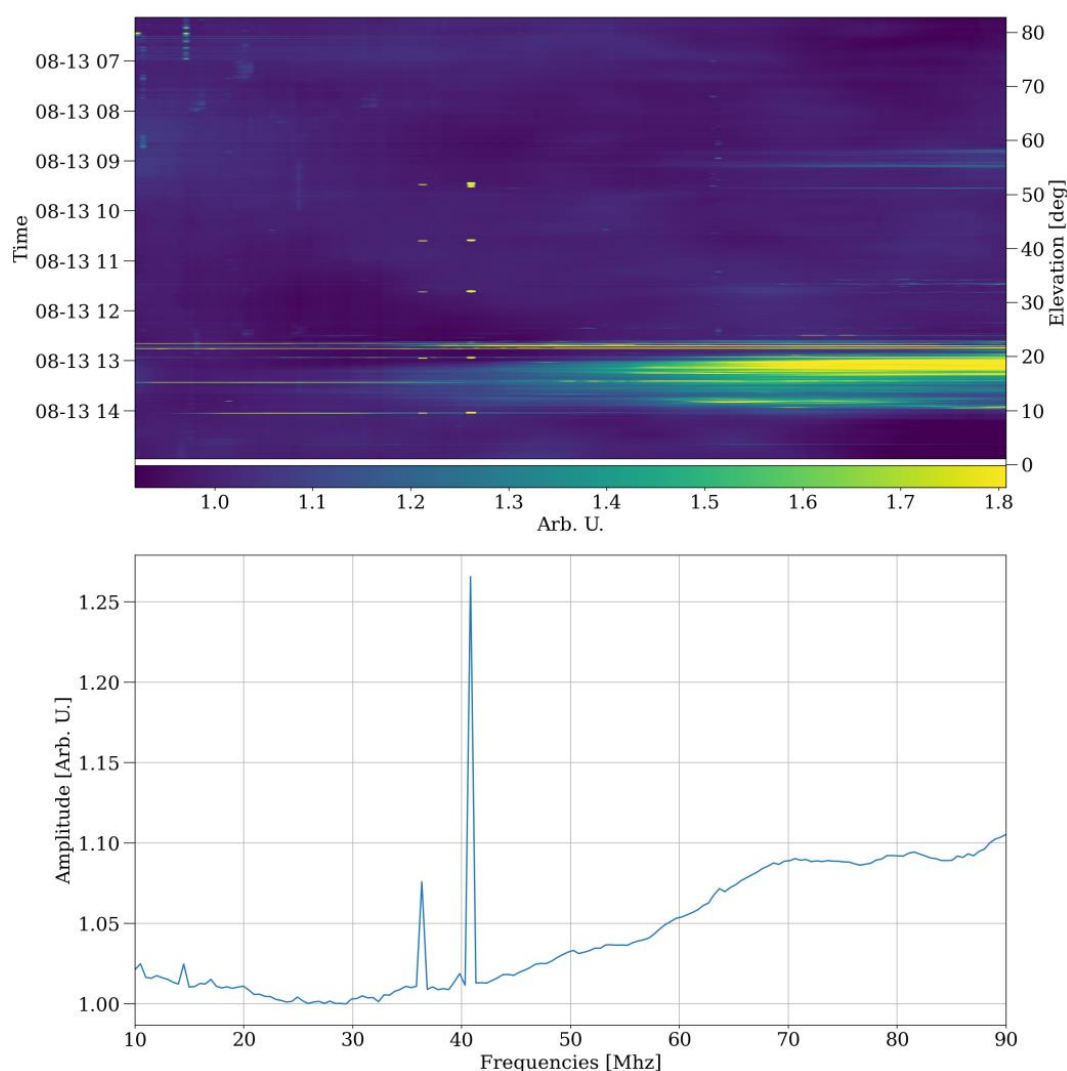
4.2. Mediānas filtrs

Mediānas filtrs ir nelineārs filtrs, kas aprēķina pikseļu intensitātītes tiem pikseļiem, kuri $n \times n$ lielā logā atrodas ap centra pikseli. Tas darbojas, ņemot visas vērtības šajā logā un iegūstot mediānu, kas tiek uzlikts kā centra pikselis. Mediānas filtrus pielieto, kad ir nepieciešams noņemt ekstrēmas vērtības, nemainot vispārējo datu struktūru, piemēram, malas. [10]

Vispirms mediānas filtrs tika pielietots visam attēlam vienlaicīgi. Tas tika realizēts, izmantojot *SciPy* pakotnes *median_filter* funkciju. Loga lielums tika izvēlēts kā 3*3 pikseļi, jo tas ir mazākais izmērs, kas deva labus rezultātus, nemainot derīgo signālu struktūru.

Tika izmēģināti dažādi filtru režīmi un salīdzinātas to veikspējas (skat. 2. tabulu). Filtru režīmi ietekmē malu pikseļu vērtību aprēķināšanu, jo logs pārsniedz attēla malu izmērus un papildus vērtības ārpus attēlam neeksistē. Šim iemeslam tiek izmantotas jau attēlā atrodošās vērtības un, norādot filtra režīmu, iespējams izvēlēties kāda tipa vērtības tiks izmantotas.

Kā redzams 5. attēlā izteikti RFI ir pamanāmi ap 35 MHz un 41 MHz frekvencēm un mazāk izteikti ap 11, 15 un 61 MHz.



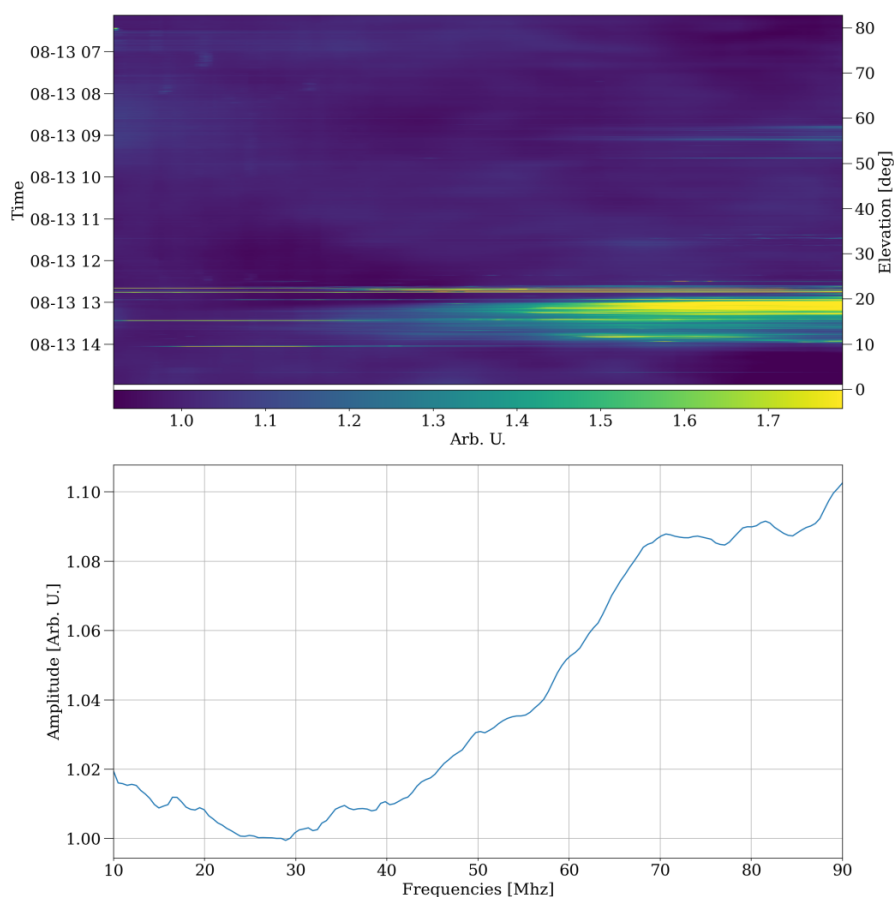
5. att. Saules (2022-08-13_06:07:39 ar novērojuma laiku 31800 sekundes) spektrs pirms mediānas filtra pielietošanas RFI mazināšanai.

Analizējot 2. tabulu var secināt, ka pēc RMSE kļūdas metrikas vislabākie rezultāti bija gan ar “reflect”, gan “nearest” mediānas filtra režīmu. Šis pats atspoguļojas arī signāla un trokšņa savstarpējā attiecībā. Tāpēc turpmākai datu apstrādei tika izvēlēts viens no tiem – “nearest” režīms.

2. tabula. *SciPy* pakotnes *median_filter* režīmu veiktspējas

Veiktspējasnr.	Režīms	RMSE kļūda	PSNR signāla–trokšņa attiecība
1.	“nearest”	0.005135	45.789945
2.	“reflect”	0.005135	45.789945
3.	“wrap”	0.005312	45.494141
4.	“mirror”	0.005343	45.444881
5.	“constant”	0.012256	38.232928

Pēc efektīvākā režīma pielietošanas attēlā iepriekš esošie RFI ir mazināti jeb izlīdzināti, atstājot spektra daļas, kuras var uzskatīt par Saules starojumu (starp 12 un 14 stundu) (skat. 6. att).



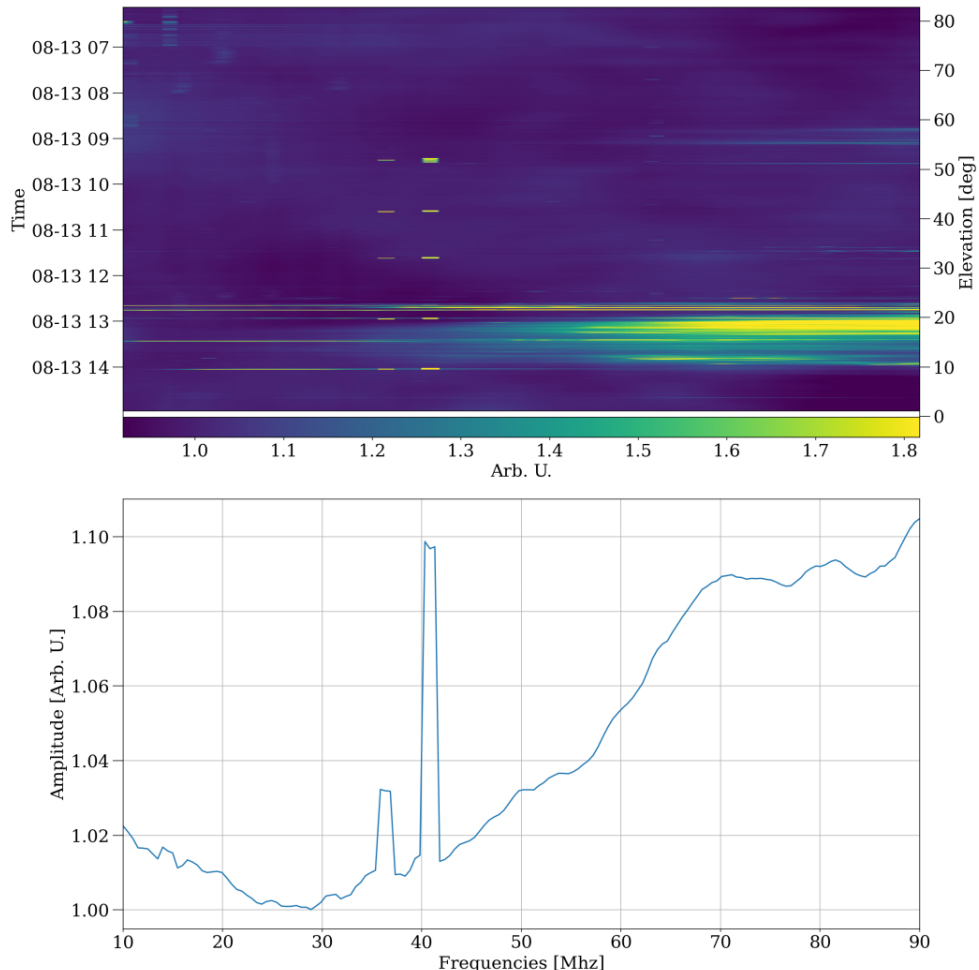
6. att. Saules (2022-08-13_06:07:39 ar novērojuma laiku 31800 sekundes) spektrs pēc mediānas filtra pielietošanas RFI mazināšanai ar “nearest” režīmu.

4.3. Vienotais filtrs

Vienotais filtrs (*angl. uniform filter*) darbojas pēc mediānas filtra principa, taču loga centrā esošā pikseļa vērtība tika iegūta, aprēķinot visu apkārt esošo loga skaitļu vidējo vērtību [11].

Testēšanai kā sākuma attēls arī tika izvēlēts 5. attēls. Filtra pielietošana tika realizēta, izmantojot *SciPy* pakotnes *uniform_filter* funkciju. Loga lielums tika izvēlēts kā 3*3 pikseļi un pielietotas “nearest” režīms, jo tas sniedza labākos rezultātus testējot mediānas filtru.

Pēc vienotā filtra pielietošanas var secināt, ka gan RFI, gan visas pārējās detaļas tika izlīdzinātas. Šis rezultāts nav apmierinošs, jo RFI netika samazināts (skat. 7. att). Papildus tam tika izpludinātas arī derīgo signālu detaļas. Var secināt, ka šī metode nesniedz apmierinošus rezultātus LOFAR attēlu apstrādes gadījumā.



7. att. Saules (2022-08-13_06:07:39 ar novērojuma laiku 31800 sekundes) spektrs pēc vienotā filtra pielietošanas RFI mazināšanai ar “nearest” režīmu

4.3. Gausa filtrs

Gausa filtrs tāpat kā mediānas filtrs samazina attēla troksni. Tas ir divdimensionāls konvolūcijas operators, kas trokšņa noņemšanu realizē, veicot attēla gludināšanu. Tas var izlīdzināt attēlus, mazinot tajos troksni, taču papildus tam samazina arī attēla smalkās detaļas. Gausa filtrs izmanto simetrisku matricu nepāra izmērā, kas tiek pārlaists katram pikselim vēlamajā attēla reģionā, lai veiktu attēla izpludināšanu. Gausa filtra darbība līdzinās mediānas un vienotajam filtram, taču vidējā pikseļa aprēķināšanai lielāku svaru dod tās vērtības, kuras ir tuvāk centra pikselim. [12]

Filtra pielietošana tika realizēta, izmantojot *SciPy* pakotnes *gaussian_filter* funkciju un testēta uz 5. attēlu. Nepieciešams izvēlēties sigmas vērtību, kas tika ņemts kā 3 un pielietots “*nearest*” režīms. Filtra centrālais pikselis tiek aprēķināts, izmantojot funkciju [12]:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad (4.1.),$$

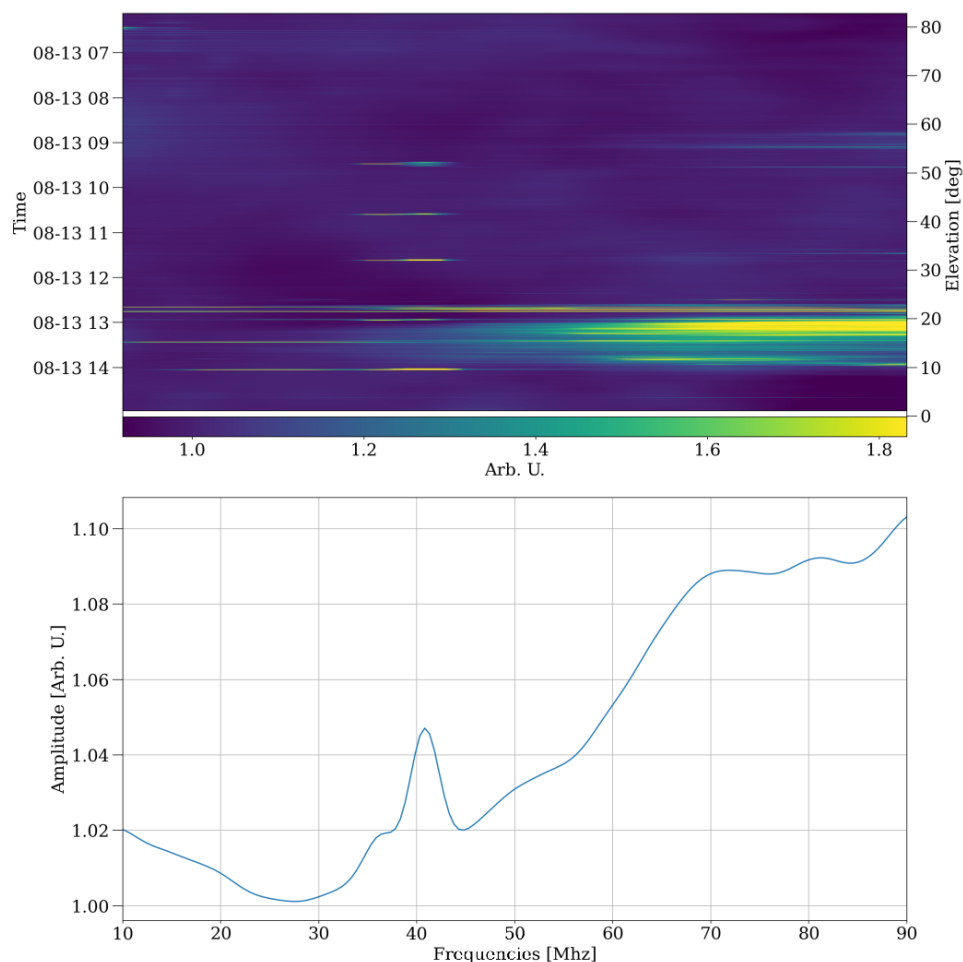
kur:

x – pikseļa x koordinātu vērtība;

y – pikseļa y koordinātu vērtība;

σ – sigma vērtība jeb izpludināšanas intensitāte.

Pēc šī filtra pielietošanas var secināt, tāpat kā ar vienoto filtru, visas attēla detaļas tika izlīdzinātas (skat 8. att).



8. att. Saules (2022-08-13_06:07:39 ar novērojuma laiku 31800 sekundes) spektrs pēc Gausa filtra pielietošanas RFI mazināšanai ar “nearest” režīmu

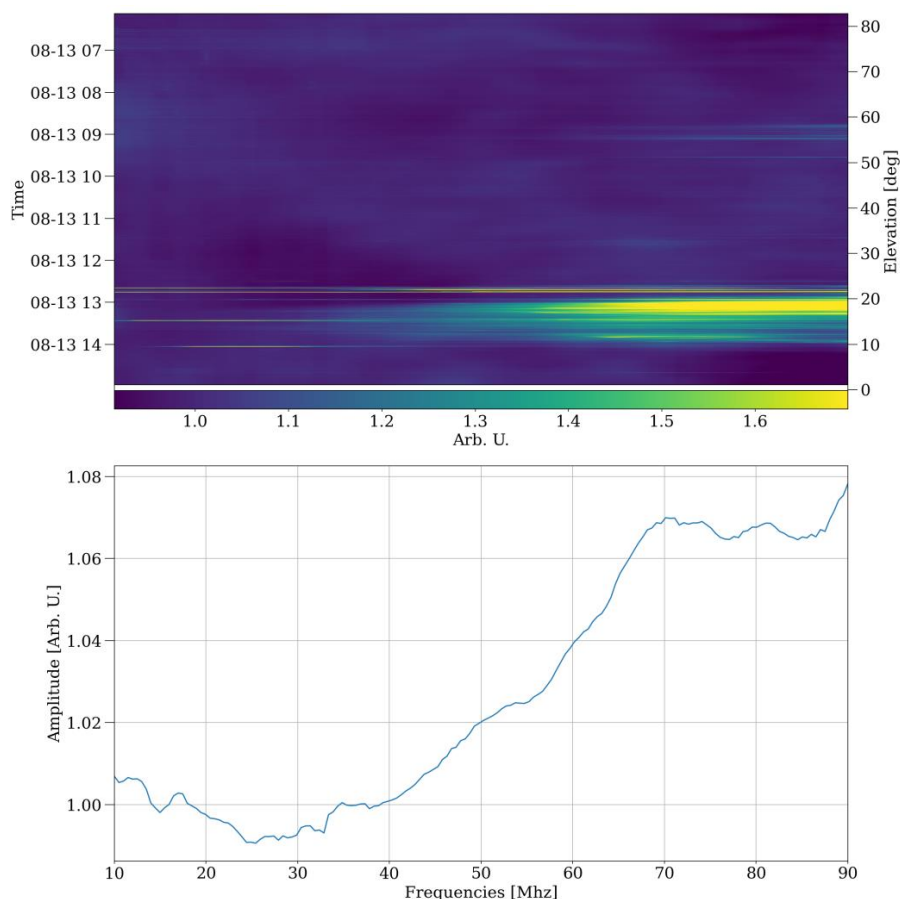
4.5. Minimuma filtrs

Minimuma filtra darbības princips ir līdzīgs visām iepriekšminētajām metodēm, taču centrālais pikselis tiek atrasts, ņemot mazāko loga vērtību [13].

Filtra pielietošana tika realizēta, izmantojot *SciPy* pakotnes *minimum_filter* funkciju un pielietots uz 5. attēla. Loga lielums tika izvēlēts kā 3*3 pikseli un pielietots “nearest” režīms.

Pētot iegūtos rezultātus (skat. 9. att.) var secināt, ka veiksmīgi noņemtas attēla nepilnības. Šajā piemērā visi RFI tika noņemti pilnībā, jo loga lielums bija pietiekoši liels, lai attēlu attīrītu no traucējumiem, bet derīgos datus atstātu gandrīz nemainīgus. Taču filtru izmantojot uz attēliem, kuros RFI atrodas lielākā laika vai frekvenču diapazonā, būtu nepieciešams palielināt loga izmēru.

SciPy pakotne satur arī maksimuma filtru, kas darbojas pretēji minimuma filtram, kur centra pikseļa vietā ņemot lielāko logā atrodošo vērtību.



9. att. Saules (2022-08-13_06:07:39 ar novērojuma laiku 31800 sekundes) spektrs pēc minimuma filtra pielietošanas RFI mazināšanai ar “nearest” režīmu

Pēc visu apskatīto filtru izmantošanas tika salīdzinātas to veikspējas, (skat. 3. tab.). Var secināt, ka vislabākos rezultātus mazu RFI noņemšanai deva minimuma filtrs.

3. tabula. Filtru veikspēju salīdzināšana

Veikspējas nr.	Filtrs	RMSE kļūda	PSNR signāla–trokšņa attiecība
1.	Minimuma	0.003911	48.153279
2.	Mediānas	0.005135	45.789945
3.	Vienotais	0.005345	45.441724
4.	Gausa	0.006708	43.468520

4.6. SumThreshold metode

SumThreshold ir iteratīva RFI noteikšanas metode, kas RFI detektē, ja kāda noteiktā attēla apakškopas vidējā vērtība ir mazāka par kādu iepriekš definētu sliekšni. Šis algoritms atrod līnijveida objektus ar izteiktām malām, kas parasti ir RFI. [15]

SumThreshold metode darbojas uz *VarThreshold* metodes principa, kas savukārt ir balstīta uz kombinatoriskās sliekšņošanas (*angl. Combinatorial thresholding*) principa. Kombinatoriskā sliekšņošanas algoritms darbojas uz principa, kur, ja kādas divas blakus esošas vērtības A un B atsevišķi nepārsniedz kādu iepriekš definētu sliekšņa vērtību T_1 , tad tiek pārbaudīts vai tie kopā pārsniedz kādu sliekšni T_2 , ja nē, tad tie var tikt apvienoti ar kādu trešo vērtību C, un kopā pārbaudīti ar trešo sliekšņa vērtību T_3 . Jo vairāk vērtības tiek kombinētas, jo zemāka sliekšņa vērtība. Algoritms ir iteratīvs un katrā iterācijā tiek mainīts vērtību kombinēšanas skaits un samazināta sliekšņošanas vērtība. [15]

VarThreshold metode papildina kombinatoriskās sliekšņošanas algoritmu ar noderīgiem parametriem, kurus būtu nepieciešams optimizēt [15]:

1. Optimālāko sliekšņa vērtību noteikšana. Jo lielāka sliekšņa vērtība, jo vairāk RFI tiks atstāti derīgajos datos, jo mazāka, jo vairāk derīgo datu tiks klasificēti kā troksnis.
2. Skaitļu sērija, kas nosaka, cik vērtības tiks kombinētas (turpmāk – logs). Skaitļi tiek kombinēti vai nu vienā frekvencē, vai laika vienībā. Logu izmēra piemērs. $M = [1, 2, 4, 8, 16, 32, \dots]$
3. Dažādas sliekšņa vērtības dažādiem logu izmēriem. To aprēķināšanai izmanto formulu:

$$T_i = \frac{T_1}{p^{\log_2 n}} \quad (4.2.),$$

kur:

T_i – sliekšņošanas vērtība iterācijā i ;

T_1 – manuāli noteikta pirmā sliekšņa vērtība;

p – empīrisks koeficients (parasti 1.5).

n – loga izmērs

4. Iterāciju skaits. Labākie rezultāti parasti tiek iegūti ap piecām iterācijām.

4.6.1. SumThershold darbības princips

SumThreshold metode uzlabo klasifikācijas veikspēju. Atšķirībā no *VarThreshold* metodes, tā summē vērtības un izmanto tās kā sliekšņošanas kritēriju, kur daļa no šīs vērtību virknes var būt mazāka par norādīto sliekšni. Pamatā tā apstrādā viendimensionālus datus, filtrējot tos ar M izmēra logu un adaptīvām sliekšņošanas vērtībām, taču divdimensionālu datu gadījumā ir iespējams apstrādāt visu attēlu, izvēloties tos apstrādāt horizontālā (frekvenču) vai vertikālā (laika) virzienā. [15]

Algoritms darbojas iteratīvi, katru iterāciju palielinot loga izmēru un samazinot sliekšņa vērtību, kur iepriekšējās iterācijās detektētie RFI turpmākajās iterācijās netiek izmantoti. [15]

Algoritma darbības princips horizontālā virzienā vienā iterācijā [14]:

1. Logs ar M izmēru tiek virzīts pāri datiem.
2. Iepriekšējā iterācijā neapskatītas vērtības tiek sasummētas (vērtību skaits virknē ir vienāds ar loga izmēru).
3. Ja vērtība pa labi iepriekšējā iterācijā nav tikusi klasificēta kā troksnis, tā tiek pieskaitīta visām pārējām logā atrodošajām vērtībām.
4. Ja vērtība pa kreisi iepriekšējā iterācijā nav tikusi klasificēta kā troksnis, tā tiek atņemta no kopējās summas.
5. Katrā loga pozīcijā vidējā vērtība var tikt aprēķināta no logā esošajām vērtībām. Ja šī vidējā vērtība pārsniedz kādu sliekšni T, tad visi logā esošās vērtības tiek klasificētas kā troksnis.

Python programmēšanas valodā realizēto kodu iespējams apskatīt pielikumā **B**. Tas balstīts uz C++ programmēšanas valodā realizētas implementācijas [16].

4.6.2. SumThreshold metodes praktiskais pielietojums

Pirms *SumThreshold* metodes pielietošanas nepieciešams veikt attēla virsmas izpludināšanu. Tiek iegūta starpība starp izpludināto un sākuma attēlu. Tas nepieciešams, lai mazinātu fona jeb derīgā signāla klasificēšanu kā troksni. [15]

Metodes testēšanai tika izmantots 5. attēls, kas ir Saules novērojums. Virsmas izpludināšanai tika veikta, izmantojot minimuma filtrēšanas metodi, jo tā iepriekšējos testos sniedza labākos rezultātus. Kā loga izmērs tika izvēlēts 11*11 pikseļi, jo tā pietiekoši veic attēla izpludināšanu un palielinot tā izmēru tiks patērēti vairāk skaitļošanas resursu, nesniedzot ievērojami labākus rezultātus.

Pēc virsmas izpludināšanas tika pielietota *SumThreshold* metode ar sekojošajiem parametriem un datiem. Filtra ievaddati ir datu matrica, kas ir starpība starp izpludināto un sākuma attēlu, kā arī maska, kas ir datu matricas izmērā un sākumā sastāv tikai no nullēm, taču ar katru iterāciju tiek papildināta ar pikseļu koordinātām, kuros noteikti RFI (klasificēta koordināta pārmainīta uz vieninieku).

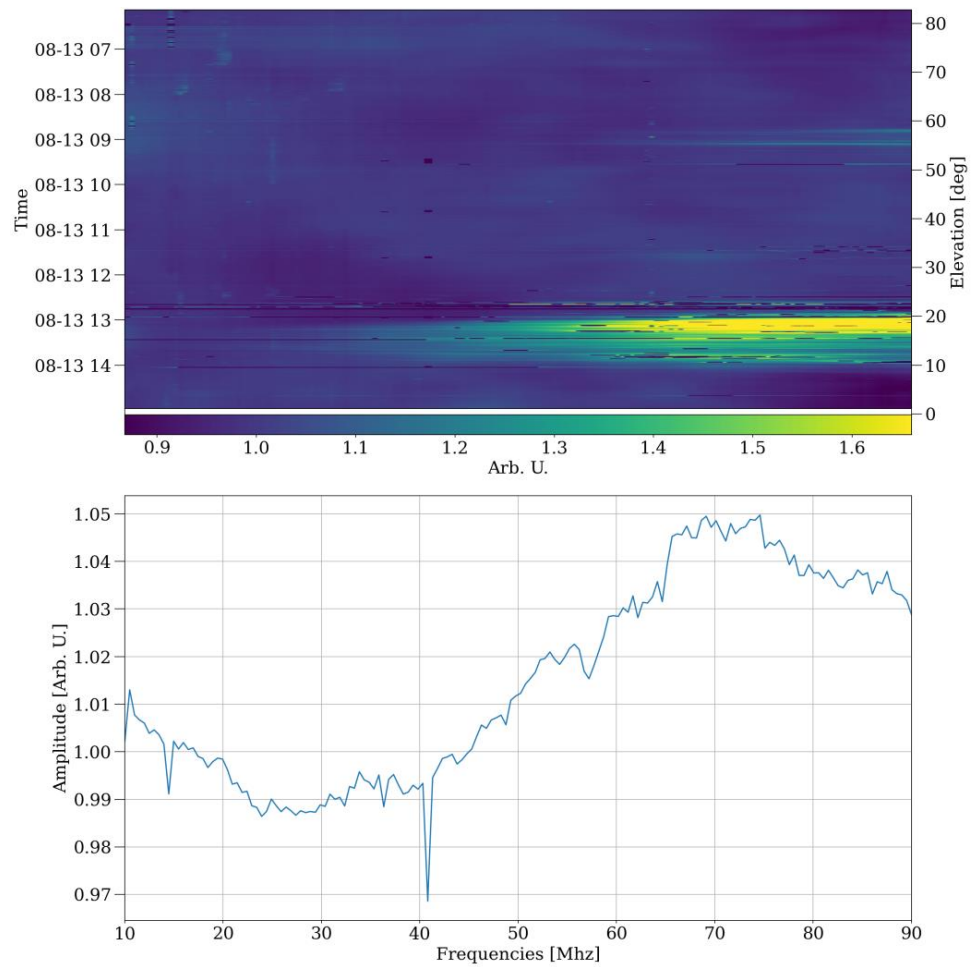
Parametri, kuri jānorāda un piemēri izmantotajam attēlam:

1. Logu izmēri: [1, 2, 4, 8, 16] (tas nozīmē, ka algoritms veiks piecas iterācijas katru iterāciju palielinot analizēto vērtību skaitu skaitļu virknē);
2. Koeficients p : 1.5;
3. Horizontālā *SumThreshold* sliekšņa vērtība: 0 (netika izmantots)
4. Vertikālā *SumThreshold* sliekšņa vērtība: 2

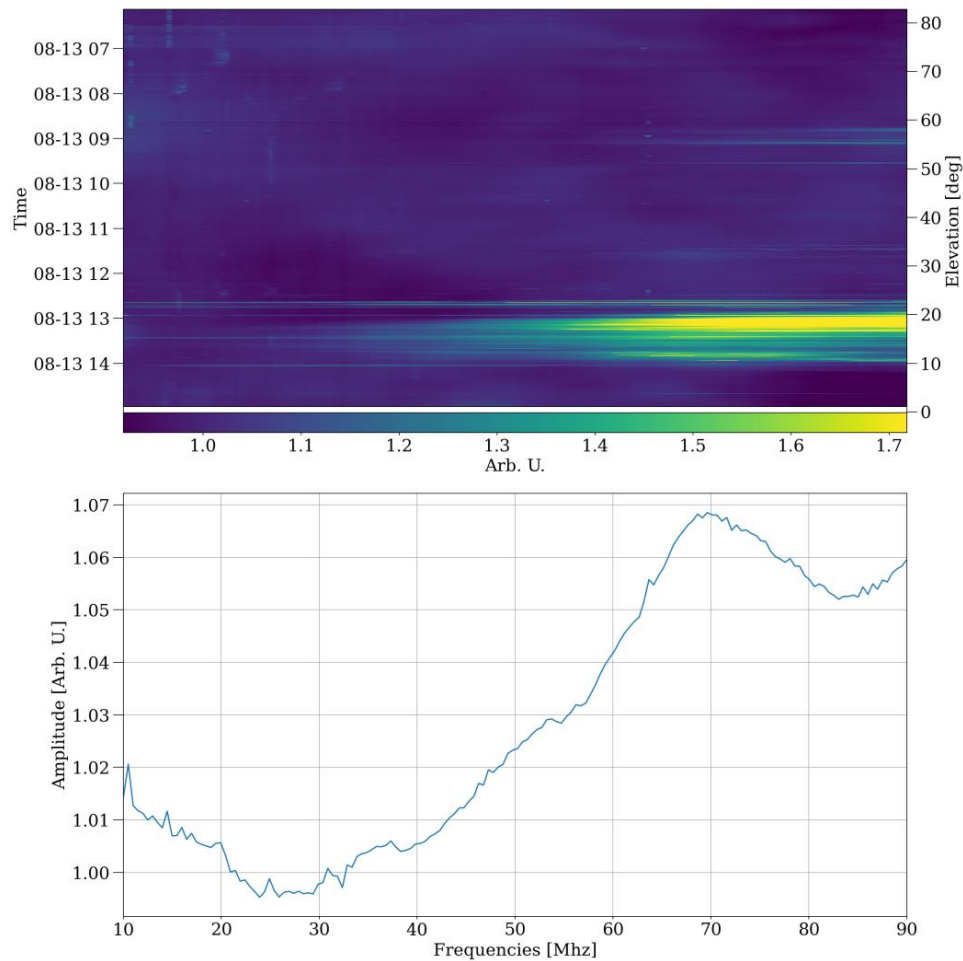
Algoritma rezultātā tiek iegūta maska ar koordinātām, kurās tika detektēts RFI. Pēc šīs maskas piemērošanas sākotnējam attēlam, šīs koordinātas oriģinālajā attēlā tika nonullētas (skat. 10. att.). Lai aizpildītu iegūtās tukšās datu vietas, tika izmantotas tuvāko pa kreisi esošo nenulles kaimiņu vērtības horizontālā virzienā. Ja šāda vērtība neeksistē, tad kā sākuma vērtība tiek ņemta attēla vidējā vērtība (skat 11. att.).

Izmantojot *SumThreshold* metodi tikai vertikālā virzienā konkrētajā attēlā tika nodrošināta optimāla RFI noņemšana, taču katram attēlam tas ir jāpielāgo individuāli, atkarībā no tajos atrodošā RFI tipa.

Rezultātā tika iegūts attēls ar samazinātu RFI (skat 11. att.). Salīdzinot 5. un 10. attēla otros grafikus, jeb frekvenču vidējās vērtības, var pamanīt, ka signālu amplitūda ir samazinājusies. Sākumā maksimālā amplitūda pārsniedza 1.25, taču pēc *SumThreshold* metodes pielietošanas tā nav lielāka par 1.05. Iegūtā attēla PSNR attiecība ir 31.45, tādējādi var secināt, ka attēlā traucējumi ir samazināti (salīdzinot ar 18.75, kas bija 10. attēlam bez nulļu vērtību aizpildīšanas).



10. att. Saules (2022-08-13_06:07:39 ar novērojuma laiku 31800 sekundes) spektrs pēc SumThreshold maskas pielietošanas, bet bez nulles vērtību aizpildīšanas.



11. att. Saules (2022-08-13_06:07:39 ar novērojuma laiku 31800 sekundes) spektrs pēc *SumThreshold* maskas pielietošanas ar nulles vērtību aizpildīšanu.

Kā redzams 10. attēlā, *SumThreshold* metode daļu no derīgā signāla klasificēja par troksni. Kaut arī nulles vērtības tika aizvietotas ar tuvāko kreiso pikseli un derīgais spektrs ir aizpildīts, tas zaudē savas sākotnējās detaļas. Tas ir tāpēc, ka *SumThreshold* metode darbojas uz principa, ka trokšņa amplitūdas ir daudz lielākas par derīgo signālu. Šo problēmu var risināt, izmantojot kalibrēšanu, kas aprakstīta 4.7. apakšnodaļā.

4.7. Kalibrēšana ar *off_target*

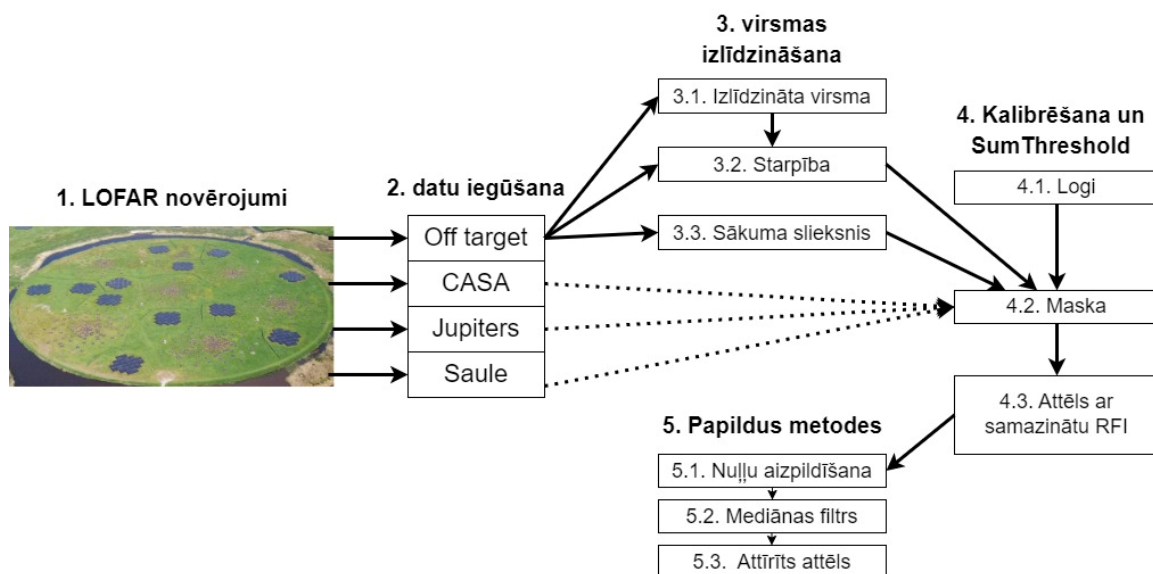
Iepriekš tika noskaidrots, ka *off_target* novērojuma mērķis ir saturēt informāciju tikai par iekšējiem RFI traucējumiem, kas būtu redzami arī visos tā paša novērojuma objektu datos. Ideālā gadījumā *off_target* nesaturētu nekādu informāciju par kosmosa objektiem. Ja šis nosacījums tiktu īstenots, tad būtu iespējams realizēt jaunu metodi RFI klasificēšanai, neietekmējot derīgos datus pārējos novērojuma objektos.

Tiek izvirzīta jauna metode, kas tiek dēvēta par “kalibrēšanu ar *off_target*”. Kā jau 4.6.2 apakšnodaļā minēts, ir nepieciešams veikt virsmas izlīdzināšanu, taču to veikšana ar attēliem, kas papildus RFI satur arī derīgos signālus ar lielām amplitūdām, kas ir tādi paši vai pat lielāki par RFI, var radīt datu zudumus un kropļojumus. Virsmas izlīdzināšana nav tieši izmantojama, ja tiek pētīti objekti, kam ir spēcīgs avots un kas rada spožu starojumu [15].

Metode darbotos veicot kalibrēšanu, sākumā pētīt un analizējot tikai *off_target* datus. No tā manuāli tiktu noteikts labākais sliekšnis (parasti tā ir lielāka vērtība par lielāko signāla amplitūdu attēlā, jo iteratīvi tā tiks samazināta) un no *off_target* datiem iegūta izlīdzinātās virsmas un sākotnējā attēla starpība, tādējādi garantējot, ka pikseļi, kuri satur lielas amplitūdu vērtības, ir tikai traucējumi.

Tālāk *SumThreshold* tiktu izmantota ar iegūtajiem virsmas izlīdzināšanas datiem no *off_target* un izvēlēto sliekšni, taču iegūtā maska tiktu piemērota jau derīgajam pētāmajam objektam. Tādējādi tiks nodrošināts, ka derīgie signāli netiktu klasificēti kā RFI.

Kalibrēšanu ar *off_target*, izmantojot *SumThreshold* metodi var sadalīt pa šādiem soļiem (skat 12. att.)



12. att. RFI mazināšanas princips LOFAR datos, izmantojot kalibrēšanu ar *off_target*. Attēla daļa “1. LOFAR novērojumi” iegūti no [2].

Soļu skaidrojums:

1. LOFAR novērojumi – veikti novērojumi un iegūti *.dat* tipa faili, kas satur datus *bst* formātā
2. Datu iegūšana no *.dat* failiem un attēlu izveidošana. Tiek iegūti dati par trīs objektiem – CASA, Jupiteru un Sauli, papildus iegūstot bez-objektu kosmosa daļas novērojumu RFI detektēšanai (*off_target*)

3. Virsmas izlīdzināšana *off_target* attēlam:
 - 3.1. Attēla izpludināšana, izmantojot minimuma filtru.
 - 3.2. Starpība starp izpludināto un oriģinālo *off_target* attēlu.
 - 3.3. Sākuma sliekšņa manuāla noteikšana, pētot oriģinālo *off_target* attēlu.
4. Kalibrēšana un *SumThreshold* metodes pielietošana:
 - 4.1. Logu skaita izvēle, kas ietekmēs iterāciju skaitu.
 - 4.2. Iegūtās maskas piemērošana kādam no objektiem jeb kalibrēšana.
 - 4.3. Rezultējošs attēls, kuram veikta RFI noņemšana, trokšņa pikselus aizstājot ar nulles vērtībām.
5. Papildus metožu pielietošana tālākai RFI mazināšanai:
 - 5.1. Nulles vērtību aizpildīšana, izmantojot tuvāko kreiso kaimiņa pikseļa vērtību horizontālā virzienā, nevarēšanas gadījumā aizpildot ar attēla vidējo vērtību.
 - 5.2. Mediānas filtra pielietošana nepieciešamības gadījumā.
 - 5.3. Galā tiek iegūts no RFI attīrīts attēls.

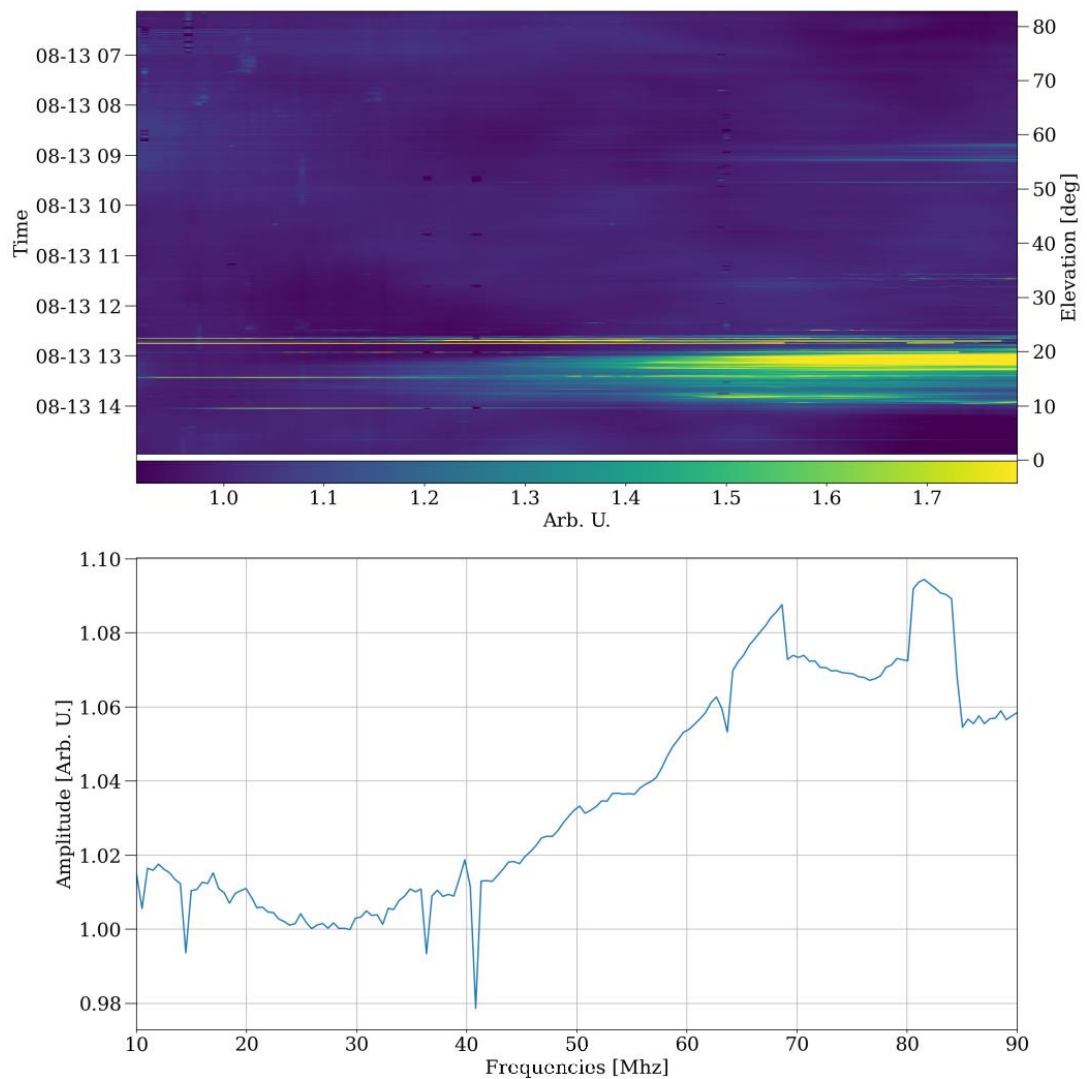
Kalibrēšana ar *off_target* darbojas tikai gadījumos, kad RFI profils sakrīt starp *off_target* un pētāmā objekta attēlu. Praktiski kalibrēšana tika veikta uz 5. attēlu, kura rezultāts ir redzams 13. attēlā. Loga izmērs tika izmantots tāds pats kā iepriekš – [1, 2, 4, 8, 16], bet sākotnējā sliekšņa vērtība tika uzstādīta kā 1. Sliekšņu vērtības samazinājās piecās iterācijās, kas tika aprēķinātas ar formulu 4.2.

Sliekšņa vērtības:

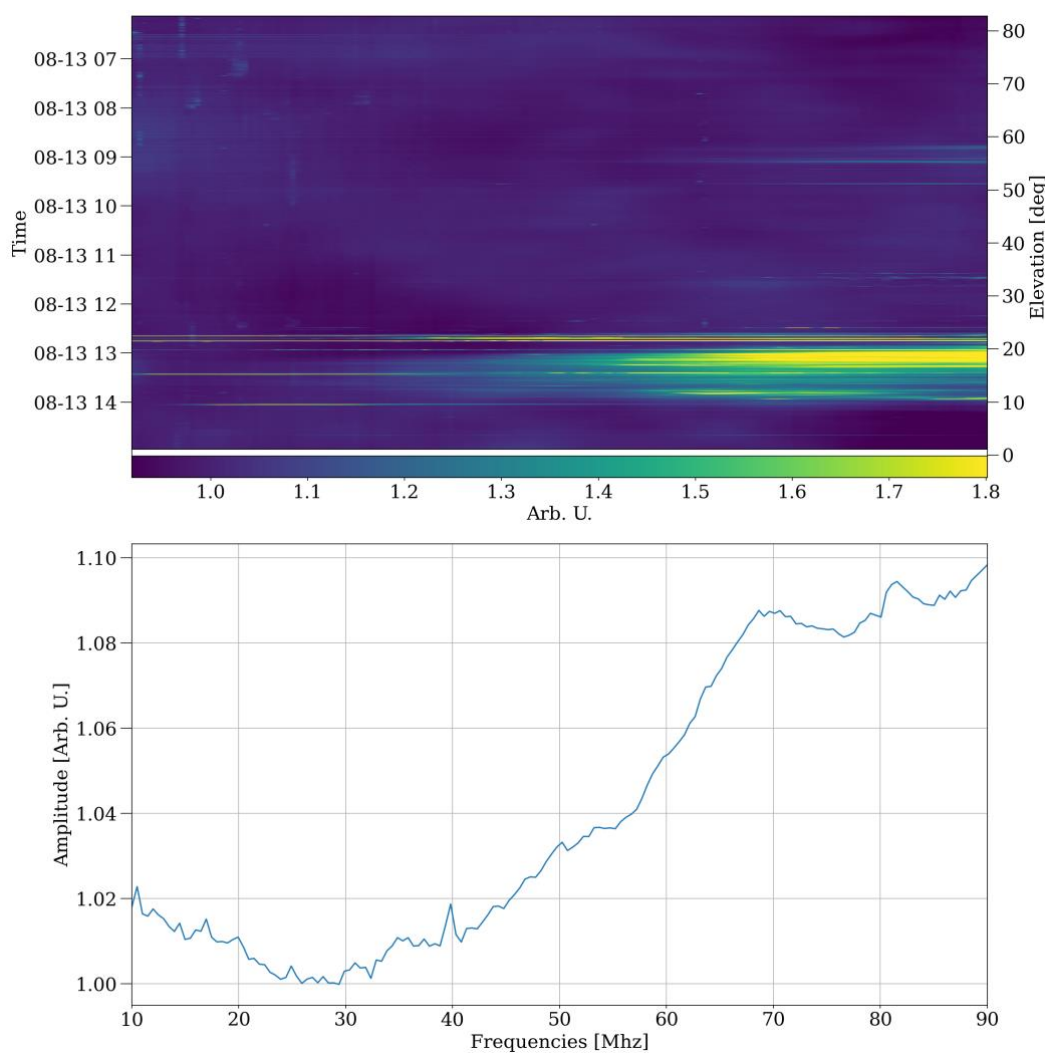
1. iterācija: 1
2. iterācija: 0.6667
3. iterācija: 0.4444
4. iterācija: 0.2962
5. iterācija: 0.1975

Var pamanīt, ka sliekšņošana ir bijusi veiksmīgāka par rezultātu, kas iegūts bez kalibrēšanas, taču pētot vidējo vērtību grafiku (13. att. apakšējais grafiks) ir redzams, ka papildus RFI tika noņemts arī derīgais signāls starp 69 un 81 MHz, kā arī frekvences pēc 84 MHz. Tas ir tāpēc, ka šajā gadījumā *off_target* attēls tomēr saturēja kādu kosmosa objektu novērojumu (skat 2. att.), kā rezultātā arī daļa no tiem tika klasificēti kā RFI un pievienoti maskai.

Pēc nulles vērtību aizpildīšanas ar tuvāko kreiso pikseli, tika iegūts spektrs ar samazinātu RFI (skat 14. att). Iegūtā gala attēla PSNR attiecība ir 53.597, kas ir lielāka par rezultātu, kas tika iegūts bez kalibrēšanas.



13. att. Saules (2022-08-13_06:07:39 ar novērojuma laiku 31800 sekundes) spektrs pēc SumThreshold maskas pielietošanas ar *off_target* kalibrēšanu, bet bez nulles vērtību aizpildīšanas.



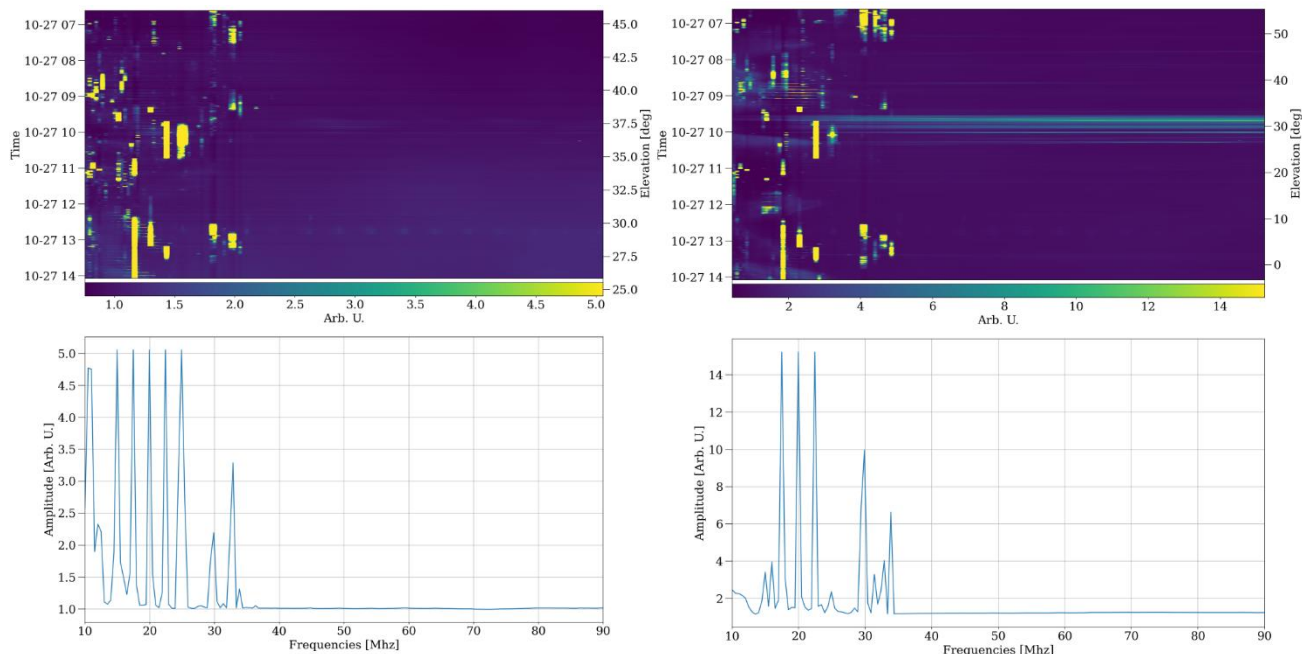
14. att. Saules (2022-08-13_06:07:39 ar novērojuma laiku 31800 sekundes) spektrs pēc SumThreshold maskas pielietošanas ar *off_target* kalibrēšanu ar nulles vērtību aizpildīšanu.

Salīdzinot gan tikai minimuma filtra veikspēju, gan *SumThreshold* metodi ar un bez kalibrēšanas, var secināt, ka *SumThreshold* metode ar kalibrēšanu deva vislabākos rezultātus, jo tika mazināts RFI, nemainot derīgo starojumu struktūru (skat. 4. tabulu.).

4. tabula. Metožu veikspēju salīdzināšana

Veiktspējas nr.	Metode	RMSE kļūda	PSNR signāla–trokšņa attiecība
1.	SumThreshold ar kalibrēšanu	0.00209	53.597
2.	<i>Minimuma</i> filtrs	0.003911	48.15
3.	<i>SumThreshold</i> bez kalibrēšanas	0.026769	31.45

SumThreshold metode ar kalibrēšanu tika izmantota arī novērojumam, kuram *off_target* attēls nesatur fona signālus, kas iegūti no kādiem kosmosa objektiem (skat 15. att. pa kreisi) un ar to tika apstrādāts Saules attēls, kas satur kādu starojumu (skat 15. att. pa labi).



15. att. *Off_target* (pa kreisi) un Saules (pa labi) (2022-10-27_06:36:47 ar novērojuma laiku 29700 sekundes) novērojumu spektrs pirms apstrādes.

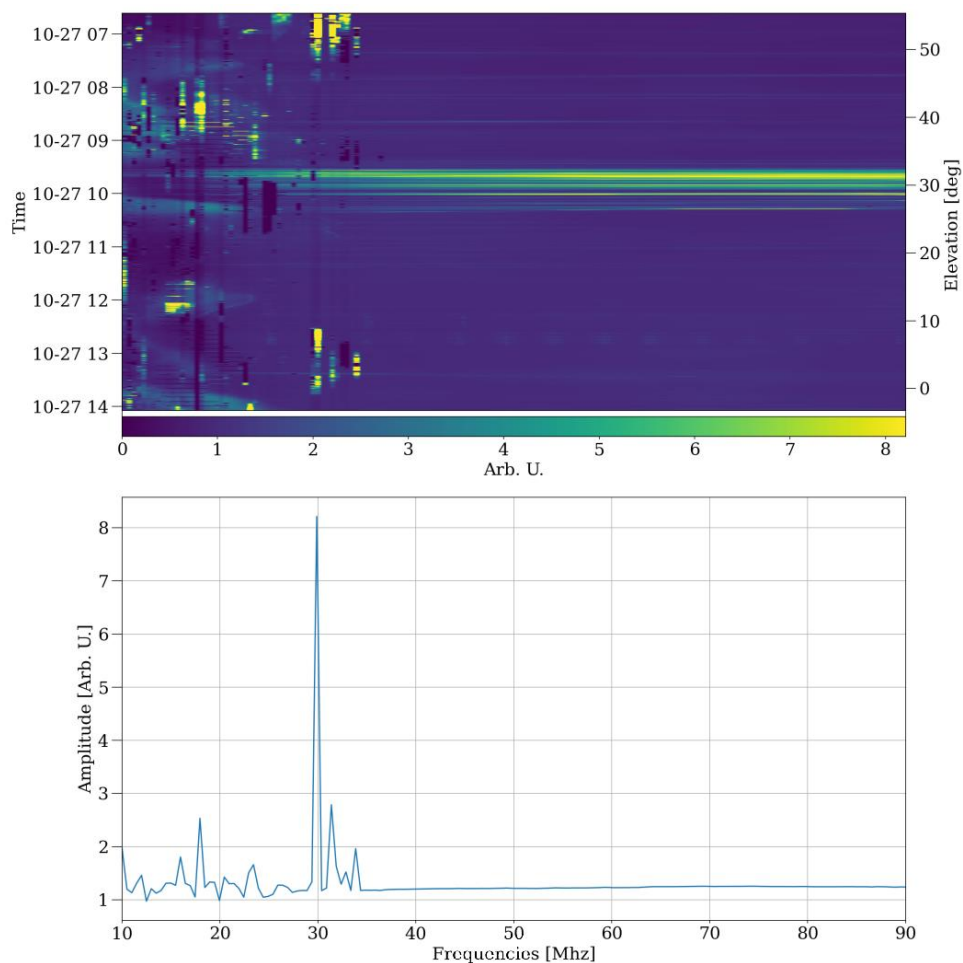
Vispirms tika norādīti parametri:

1. Logu izmēri: [1, 2, 4, 8, 16]
2. Koeficients p : 1.5;
3. Horizontālā *SumThreshold* sliekšņa vērtība: 0 (netika izmantots)
4. Vertikālā *SumThreshold* sliekšņa vērtība: 4

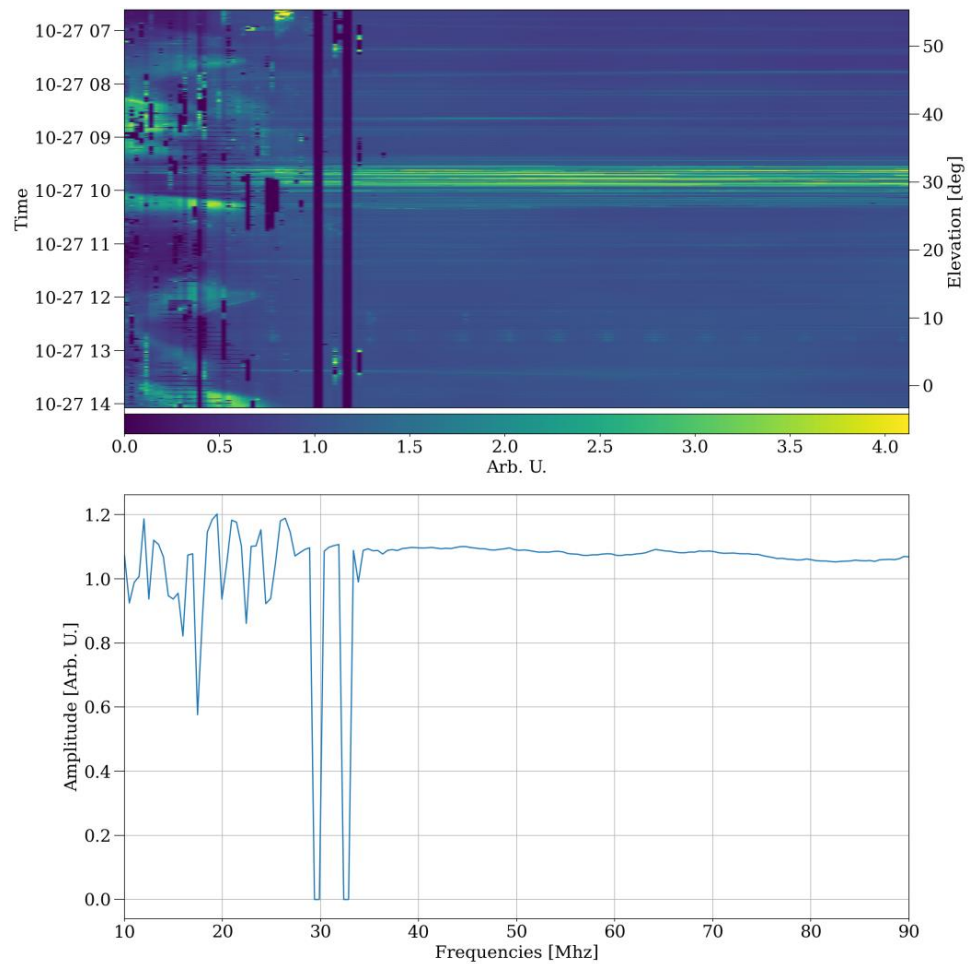
Tā kā šajā novērojumā *off_target* un Saules attēlos atrodošo RFI profili atšķīrās, tad nebija iespējams veikt pilnīgu RFI noņemšanu, izmantojot tikai kalibrēšanu (skat 16. att.). Šim iemeslam *SumThreshold* metode tika izmantota vēlreiz uz kalibrētajiem datiem, taču jauno masku aprēķinot bez kalibrēšanas, lai būtu iespējams apstrādāt Saules attēla atrodošo RFI profilu. Tika mainīti norādītie parametri:

1. Logu izmēri: [1, 2, 4]
2. Vertikālā *SumThreshold* sliekšņa vērtība: 8

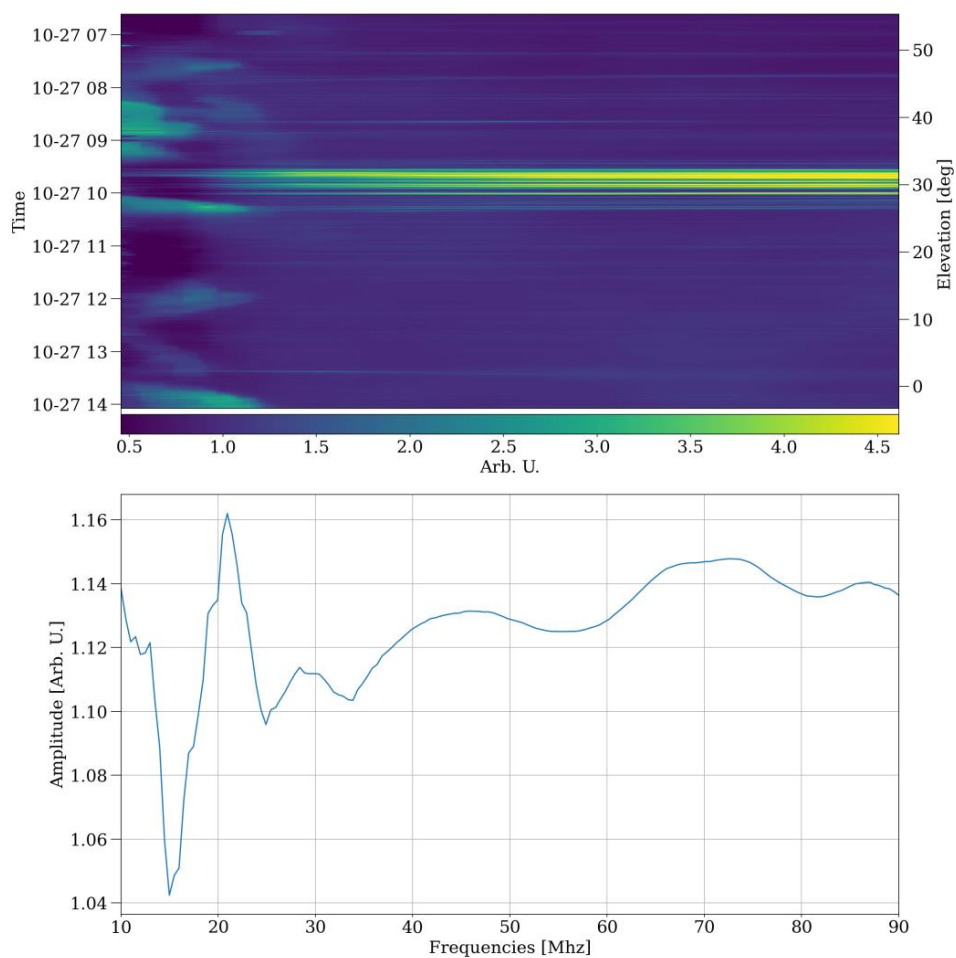
Iegūtais rezultāts redzams 17. attēlā. Otrajā grafikā redzams, ka RFI amplitūda ir samazinājusies. Sākotnēji 16. attēlā redzams, ka trokšņa vidējā amplitūda 30 MHz frekvencē pārsniedz 8, bet pēc otras SumThreshold metodes pielietošanas signāla vidējā amplitūda nepārsniedz 1.2, RFI attēlā ir mazināts. Pēc tam tika pielietota nulles vērtību aizpildīšana ar tuvāko kreiso pikseli. Tika secināts, ka attēls vēl ir trokšņains, tāpēc tika pielietots mediānas filtrs ar loga izmēru 11*11 (skat 18. att.), lai veiktu pēdējo trokšņu noņemšanu. Rezultātā iegūts no RFI attīrīts attēls, taču ar mazliet izmainītām derīgo signālu vērtībām, mediānas filtra pielietošanas dēļ.



16. att. Saules (2022-10-27_06:36:47 ar novērojuma laiku 29700 sekundes) novērojumu spektrs pēc *SumThreshold* metodes pielietošanas ar *off_target* kalibrēšanu.



17. att. Saules (2022-10-27_06:36:47 ar novērojuma laiku 29700 sekundes) novērojumu spektrs pēc vēl vienas *SumThreshold* sliekšņošanas metodes izmantošanas.



18. att. Saules (2022-10-27_06:36:47 ar novērojuma laiku 29700 sekundes) novērojumu spektrs pēc nulles vērtību aizpildīšanas un mediānas filtra pielietošanas ar loga izmēru 11*11.

5. SECINĀJUMI UN PRIEKŠLIKUMI

SECINĀJUMI:

1. RFI attēlos var klasificēt pēc to veidiem, kur katram no tiem vislabākos rezultātus deva cita apskatītā metode. Ne vienmēr nepieciešams izmantot *SumThreshold* metodi, ja iespējams RFI mazināšanu veikt ar kādu no parastajiem filtriem (skat. 4. tabulu).

4. tabula. RFI veidi un labākās metodes to mazināšanai.

RFI veids	Metode, kas sniedza labākos rezultātus	Papildnosacījumi / Tūkumi
Mazi RFI, kas ir periodiski laika virzienā un kuru izmērs nepārsniedz dažus pikseļus (skat 5. att.)	1. Mediānas filtrs	Mazliet izmainītas derīgā signāla detaļas
Mazi RFI, kas ir periodiski laika virzienā un kuru izmērs nepārsniedz dažus pikseļus (skat 5. att.)	2. Minimuma filtrs	Papildus RFI zūd derīgā signāla detaļas
Lieli RFI, kas bojā datus plašākā laika vai frekvenču diapazonā (skat. 14. att.)	<i>SumThreshold</i> metode vertikālā vai horizontālā virzienā	RFI nesakrīt starp <i>off_target</i> un objekta attēlu
Lieli RFI, kas bojā datus plašākā laika vai frekvenču diapazonā (skat. 14. att.)	<i>SumThreshold</i> metode vertikālā vai horizontālā virzienā ar <i>off_target</i> kalibrēšanu	RFI sakrīt starp <i>off_target</i> un objekta attēlu

2. Lai LOFAR attēlus attīrītu no lieliem RFI, nepieciešams izmantot metožu kopumu. *SumThreshold* metodes pielietošanas gadījumā tā ir virsmas izpludināšana ar minimuma filtru, sliekšņošana ar *SumThreshold*, nulles vērtību aizpildīšana un nepieciešamības gadījumā atlikušo RFI noņemšana ar mediānas filtru.
3. Parastā sliekšņošana var būt derīga tikai tiem attēliem, kuri satur izteiktu RFI kādā noteiktā frekvencē, kas amplitūdās ir daudz lielāks par derīgajiem signāliem. Uz LOFAR attēliem tas, galvenokārt, nav attiecināms, jo tie var saturēt derīgu signālu un RFI komponentes ar vienādām intensitāšu amplitūdām.
4. Mediānas filtrs labi darbojas uz attēla reģioniem, kuri satur mazus RFI, pretējā gadījumā lielas RFI daļas tiek izpludinātas. Šādā gadījumā tā jālieto kopā ar *SumThreshold* metodi.
5. Minimuma filtrs, tāpat kā mediānas filtrs, arī labi darbojas mazu RFI traucējumu noņemšanai, ja tie nav lielāki par dažiem pikseļiem, taču tas izmaina arī derīgā signāla struktūru.

6. Gausa filtrs labāk attīra attēlus, kas satur vienādu troksni visā attēlā, piemēram, Gausa troksni, taču neattīra attēlos no trokšņiem, kuri atrodas plašākā laika vai frekvenču diapazonā. LOFAR attēlu gadījumā tas RFI izpludina, taču no attēla nenņem. Līdzīgi attēlu izmaina vienotais filtrs.
7. Jo lielāks *SumThreshold* sākuma sliekšnis, jo vairāk RFI tiek atstāti attēlā. Turpretī, jo tas ir mazāks, jo vairāk derīgie signāli tiks klasificēti kā RFI.
8. Ja horizontālā un vertikālā virzienā izmanto vienu un to pašu sliekšni vai loga lielumus, tad rezultāti ne vienmēr būs korekti, jo RFI tiks uzskatītas arī derīgā signāla daļas vai RFI daļas netiks klasificētas. Katrs LOFAR attēls ir jāanalizē atsevišķi un jānosaka, kuru sliekšņošanas virzienu (vai abus), kādu sliekšņa vērtību un kādus loga izmērus nepieciešams uz to izmantot.
9. VSRC LOFAR Latvia stacija iegūst datus, kuros RFI lielākoties ir laika domēnā, tāpēc nepieciešams pielietot *SumThreshold* vertikālā virzienā.
10. Tā kā VSRC veido objektu attēlus, kuros derīgie signāli ir frekvenču domēnā, tad *SumThreshold* metodes izmantošana horizontālā virzienā ir jāveic uzmanīgi, lai derīgie signāli netiktu klasificēti kā troksnis. Tas nozīmē, ka algoritms horizontālā virzienā jāveic mazākā iterāciju skaitā un jāizmanto lielāka sākuma sliekšņa vērtība.
11. *SumThreshold* metode darbojas pēc principa, ka derīgais signāls vidēji ir mazāk intensīvs par traucējumiem, taču pielāgojot loga izmērus, sākuma sliekšņa vērtības un izmantojot papildus metodes, iespējams iegūt apmierinošus attīrīšanas rezultātus.
12. Lai novērstu derīgo signālu klasificēšanu kā RFI, nepieciešams veikt kalibrēšanu ar *off_target* attēlu, kur tam ir jāsaturs tikai informācija par RFI.
13. Ja pēc kalibrēšanas ar *off_target* tajā atrodošais RFI profils ir atšķirīgs no RFI profila pētāmajā objekta attēlā, tad RFI klasificēšana un noņemšana nebūs korekta. RFI koordinātām jeb struktūrai *off_target* un objekta attēlos ir jāsakrīt.

PRIEKŠLIKUMI:

1. Pēc *SumThreshold* metodes pielietot skalas invarianta rangs jeb SIR (*angl. Scale Invariant Rank*) operatoru, kas paplašinātu pikseļu klasificēšanu ārpus noteiktajām RFI robežām, jo reizēm klasificētajiem RFI pikseļiem blakus esošās vērtības arī satur traucējumus.
2. Ieviest sākotnējo sliekšņa vērtību noteikšanas automatizēšanu, lai tas nebūtu jānosaka manuāli katram attēlam.

3. Automatizēt RFI salīdzināšanu starp objekta un *off_target* attēlu, lai varētu noteikt, vai iespējams pielietot kalibrēšanu.
4. Pēc maskas pielietošanas iegūtās nulles vērtības aizpildīt, izmantojot kādu citu aizpildīšanas metodi
5. Izmantot ģenerētus LOFAR attēlus, lai varētu kontrolēt RFI daudzumu un veikt testēšanu.
6. RFI noteikšanai pielietot mašīnmācīšanās algoritmus, kas ir apmācīti tieši uz LOFAR datiem.

IZMANTOTĀS LITERATŪRAS UN AVOTU SARAKSTS

- [1] *LOFAR telescope* [skatīts 2022. gada 4. decembrī]
Pieejams: www.astron.nl/telescopes/lofar/
- [2] van Haarlem, Michael P., Michael W. Wise, A. W. Gunst, George Heald, John P. McKean, Jason WT Hessels, A. Ger de Bruyn et al. "LOFAR: The low-frequency array." *Astronomy & astrophysics* 556 (2013): A2.
- [3] An, Tao, Xiao Chen, P. Mohan, and Bao-Qiang Lao. "Radio frequency interference mitigation." arXiv preprint arXiv:1711.01978 (2017).
- [4] *RFI filter basics* [skatīts 2023. gada 23. janvārī]
Pieejams: www.astrodynetdi.com/blog/rfi-filter-basics
- [5] *LOFAR station data cookbook* [skatīts 2023. gada 28. janvārī]
Pieejams: lofar.ie/wp-content/uploads/2018/03/station_data_cookbook_v1.2.pdf
- [6] Shweta Gupta, *RMSE: What does it mean?* [skatīts 2023. gada 23. janvārī]
Pieejams: medium.com/@mygreatlearning/rmse-what-does-it-mean-2d446c0b1d0e
- [7] Cadence PCB Solutions, *What is Signal to Noise Ratio and How to calculate it?* [skatīts 2023. gada 23. janvārī]
Pieejams: resources.pcb.cadence.com/blog/2020-what-is-signal-to-noise-ratio-and-how-to-calculate-it
- [8] *Signal-to-noise Ratio* [skatīts 2023. gada 24. janvārī]
Pieejams: deepai.org/machine-learning-glossary-and-terms/signal-to-noise-ratio
- [9] Tamilselvan, Kumaravel Subramaniam, and Govindasamy Murugesan. "Image Segmentation." *Medical and Biological Image Analysis* 1 (2018).
- [10] *Median filter* [skatīts 2023. gada 24. janvārī]
Pieejams: www.sciencedirect.com/topics/computer-science/median-filter
- [11] *Mean filter* [skatīts 2023. gada 24. janvārī]
Pieejams: homepages.inf.ed.ac.uk/rbf/HIPR2/mean.htm
- [12] *Apply a Gauss filter to an image with Python* [skatīts 2023. gada 27. janvārī]
Pieejams: www.geeksforgeeks.org/apply-a-gauss-filter-to-an-image-with-python/
- [13] *Minimum filter:* [skatīts 2023. gada 27. janvārī]
Pieejams: www.digimizer.com/manual/m-image-filtermin.php
- [14] Offringa, André. "The SumThreshold method: technical details." Tech. Rep. (2012).
- [15] Offringa, A. R., A. G. De Bruyn, M. Biehl, S. Zaroubi, G. Bernardi, and V. N. Pandey. "Post-correlation radio frequency interference classification methods." *Monthly Notices of the Royal Astronomical Society* 405, no. 1 (2010): 155-167.

- [16] André Offringa, *SumThreshold algoritma implementācija C++ programmēšanas valodā* [skatīts 2023. gada 24. janvārī] Pieejams: gitlab.com/aroffringa/aoflagger/-/blob/master/algorithms/sumthreshold.cpp

PIELIKUMI

A .sh faila piemērs LOFAR novērojumu veikšanai

```
rcumode=3
duration=31800 # observation duration in seconds
observation='sun_TEST'
today=""date +"%Y.%m.%d""
start_time=""date +"%H%M%S""
datapath="/mnt/LOFAR1/ON_OFF_TEST/'$today'/'$observation'/'$start_time'_rcumode'$rcumode
echo 'Datapath = '$datapath
mkdir -p $datapath

cp obs_sun.sh $datapath

antennaset='LBA_INNER'
subbands='150:311'
band='10_90'

CASA="6.123487680622105,1.0265153995604648,J2000"
OFF_TARGET="6.218333453664764,0.9920527098688247,J2000"
SUN="0,0,SUN"

bits=8

echo 'Setting bitmode'
rsptcl --bitmode=$bits
sleep 10

echo 'Setting CAS A beam, '$CASA', band='$band', subbands='$subbands', antennaset='$antennaset', bits='$bits'
nohup beamctl --antennaset=$antennaset --rcus=0:191 --band=$band --subbands=$subbands --beamlets=0:161 --anadir=$CASA --digdir=$CASA > $datapath/beamctl_1.log 2>&1 &
sleep 10

echo 'Setting sun beam, '$SUN' band='$band', subbands='$subbands', antennaset='$antennaset', bits='$bits'
nohup beamctl --antennaset=$antennaset --rcus=0:191 --band=$band --subbands=$subbands --beamlets=162:323 --anadir=0,0,SUN --digdir=0,0,SUN > $datapath/beamctl_2.log 2>&1 &
sleep 10

echo 'Setting OFF beam, '$OFF_TARGET', band='$band', subbands='$subbands', antennaset='$antennaset', bits='$bits'
nohup beamctl --antennaset=$antennaset --rcus=0:191 --band=$band --subbands=$subbands --beamlets=324:485 --anadir=$OFF_TARGET --digdir=$OFF_TARGET > $datapath/beamctl_3.log 2>&1 &
sleep 10

echo 'Start recording'
nohup rsptcl --statistics=beamlet --select=0,1 --duration=$duration --integration=1 --directory=$datapath > $datapath/rsptcl_beamlet.log 2>&1
echo 'Finished recording'
echo 'Observation finished, killing the beam'
killall beamctl # kill any existing beams
```

B Izmantotās PSNR un RMSE precizitāšu noteikšanas un SumThreshold metodes funkcijas

SumThreshold metodes un precizitāšu noteikšanas funkcijas pieejamas *github* repozitorijā: github.com/Emilsrie/rfi_removal_tools