Camera Subsystem

Autotune SmartU2 User Guide

v0.5

July 6, 2020

Submit technical questions to:

AutoTune.team@qti.qualcomm.com

Qualcomm Technologies, Inc.
5775 Morehouse Drive
San Diego, CA 92121
U.S.A.

Revision history

Revision	Date	Description
0.1	20-April-2020	First Draft : Kamorta Autotune in C7
0.2	18-May-2020	Added sections on Metrics Scores and Parameters Tuned by Autotune
0.3	01-June-2020	Updates with respect to Kamorta Video flow enablement
0.4	08-June-2020	Update for De-Ghosting Strength , and Video flow C7 simulation
0.5	6-July-2020	Update for Recommended Flow and restriction on Fine Tuning with Chroma Enable

Contents

Figures

Tables

No table of figures entries found.

1 Preface

This is the user's guide for the CamSS Autotune SmartU2 for Kona/Kamorta ISPs. Autotune SmartU2 tool is integrated into the Chromatix (C7) and provides a GUI interface for reference tuning the camera ISP. The Autotune tool will tune the Noise and Sharpness related modules.

The process of executing the Autotune consists of three steps:

1. C7 Simulation (Section **C7 Simulation)** – This is the first step to run Autotune. This needs to be run every time when we are tuning for a particular sensor/lux condition. This step evaluates whether the inputs provided to Autotune are correct and prompts an error in case of a wrong input. This is important since the next step is very compute intensive and may run for longer time based on the input resolution/usecase – in case of any wrong inputs it's better to prompt the user at this step
2. Coarse Tuning – Precheck/Build (Section **Autotune Coarse Tuning - PreCheck/Build**) – This step evaluates the sensor under the given lux condition and develops a database (called "sensor database") of various image quality tradeoff points achievable by the ISP. The tool then uses this sensor database to perform an automatic coarse level tuning of the ISP with the given reference image as the target.
3. Fine Tuning (Section **Steps to Run Autotune – Fine Tune**) **-** This is an interactive step after the completion of Coarse Tuning and enables the user to further tune the modules based on his/her image quality preferences.

The Autotune SmartU2 tool can be used to tune the Noise and Sharpness module parameters. The tool assumes that a basic tuning is performed on all other modules that are not tuned by the SmartU2 tool and is provided in the form of a Chromatix7 (C7) Base project.

2 Requirements to Run Autotune

2.1 Chromatix Project and Tuning Chart Requirements

The below are the requirements to run the Autotune tool. This should be available before we start the Autotune execution:

1. Basic tuned C7 project
   - 3A, black level, color, tone mapping and any other basic tuning should be done.
   - Modules that need to be autotuned can have default settings.
   - Should have all the AEC trigger regions that need to be tuned.
2. The gain trigger regions for all modules to be tuned (by Autotune) should be aligned.

   Autotune will tune the following modules in Kamorta
   - Snapshot flow – ABF, WNR, ASF
   - MFNR flow – SW_MFNR, WNR(Postfilter), ASF
   - Video flow – WNR, ASF, SWMCTF

     Note: For Video usecase, AutoTune tunes only spatial_denoising_strength parameter. It is assumed that the blend_strength and motion_detection_sensitivity are already well tuned in the base project. Also, enableMC should be 1.
3. Static TE42 captures with the sensor
   - Tripod captures are preferable to rule out registration related issues.
   - The gain of TE42 captures should fall within the trigger regions' start and end gain values.
   - There should be one-is-to-one mapping between C7 trigger regions and TE42 captures.
   - Need TE42 captures for every trigger region that needs to be tuned.
   - Video flow also expects static TE42 captures. (It is recommended to use 10 frames for optimal tuning and execution time.)
4. A target TE42 image for each trigger region is required.
   - Instead of target image, target metrics can be provided as an xml file.
   - For Video flow, Autotune does not accept target videos. It is recommended to provide the target IQ metrics in target XML format.

2.2 Machine Requirements

1. Machine Configuration
   - Free Disk Space of 100 GB or more
   - RAM minimum of 16 GB, preferable is 64GB or 128 GB
   - CPU Cores – 16 cores minimum, 32 cores preferable

2.3 Tool Dependencies

The following two packages are required for running the Autotune tool:

- Visual C++ Redistributable for Visual Studio 2015
- Visual C++ Redistributable for Visual Studio 2017

2.4 Execution Time

The below table shows the approximate running time for each step of Autotune. Please note that the running time will depend on the Machine configuration and the resolution of the images used for tuning.

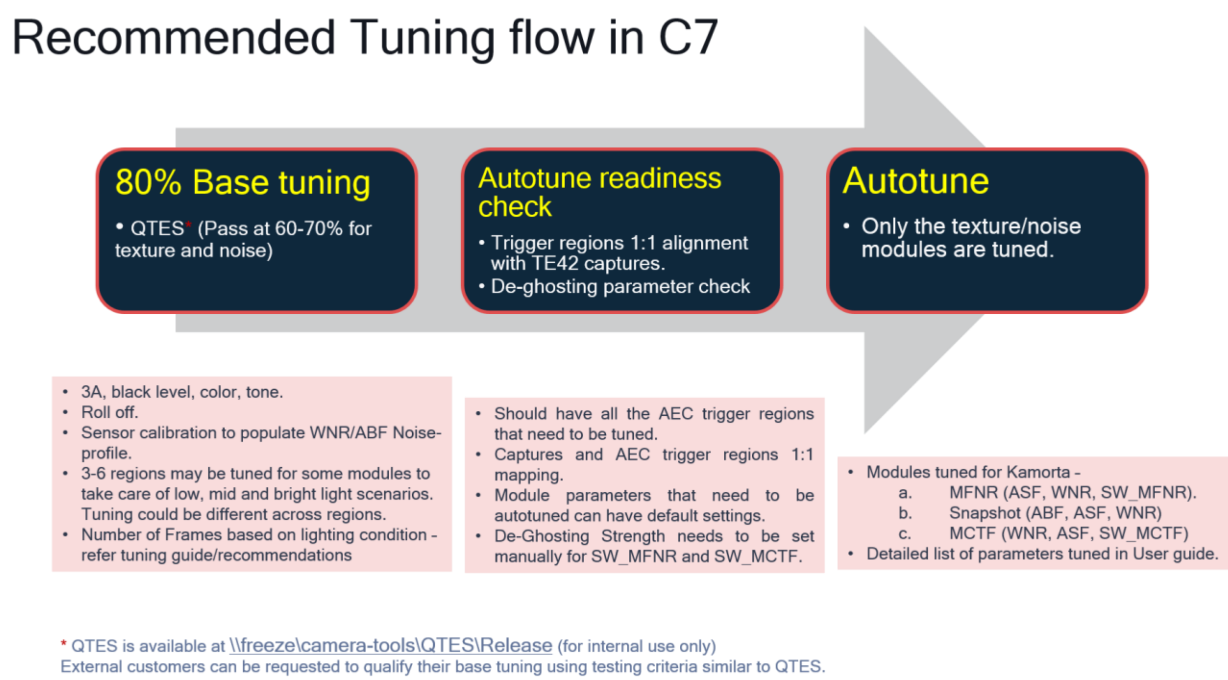
The below data is average number for multiple runs and profiled using a Windows 10 PC with 16 Cores (2 threads per core), 64GB RAM for tuning image resolution of 15 MP. Please use these as guideline average running times.

	Precheck	**Build**	**Finetune (simulation for 1 point)**
Snapshot	5 Mins	5 Hrs	5 Mins
MFNR (8 Frames)	30 Mins	12.5 Hrs	10 Mins
Video (10 frames of resolution 1920x1080)	5 mins	6 Hrs	3 Mins

3 How to Run Autotune

3.1 Recommended Tuning Flow in Chromatix

The below is the recommended flow to get the best results from Autotune



It is required that we use 80% Base Tuned project for running Autotune. Please note that Autotune is used for Noise/Texture/Resolution tuning. The sensor calibration, 3A, black level, lens roll off are to be set correctly/calibrated before running Autotune.

3.2 Autotune Flow

Figure 3-1 shows the Autotune Flow as a Flowchart. Please note that first we need to load the C7 (Chromatix) project and simulate the usecase on the C7. This will generate the spectracamsim config, which is an input to the Autotune. Also, the output generated by the C7 simulation should be correct (check the jpg in the C7 Simulation output folder). Proceed to Autotune only if you are able to simulate the C7 project correctly.

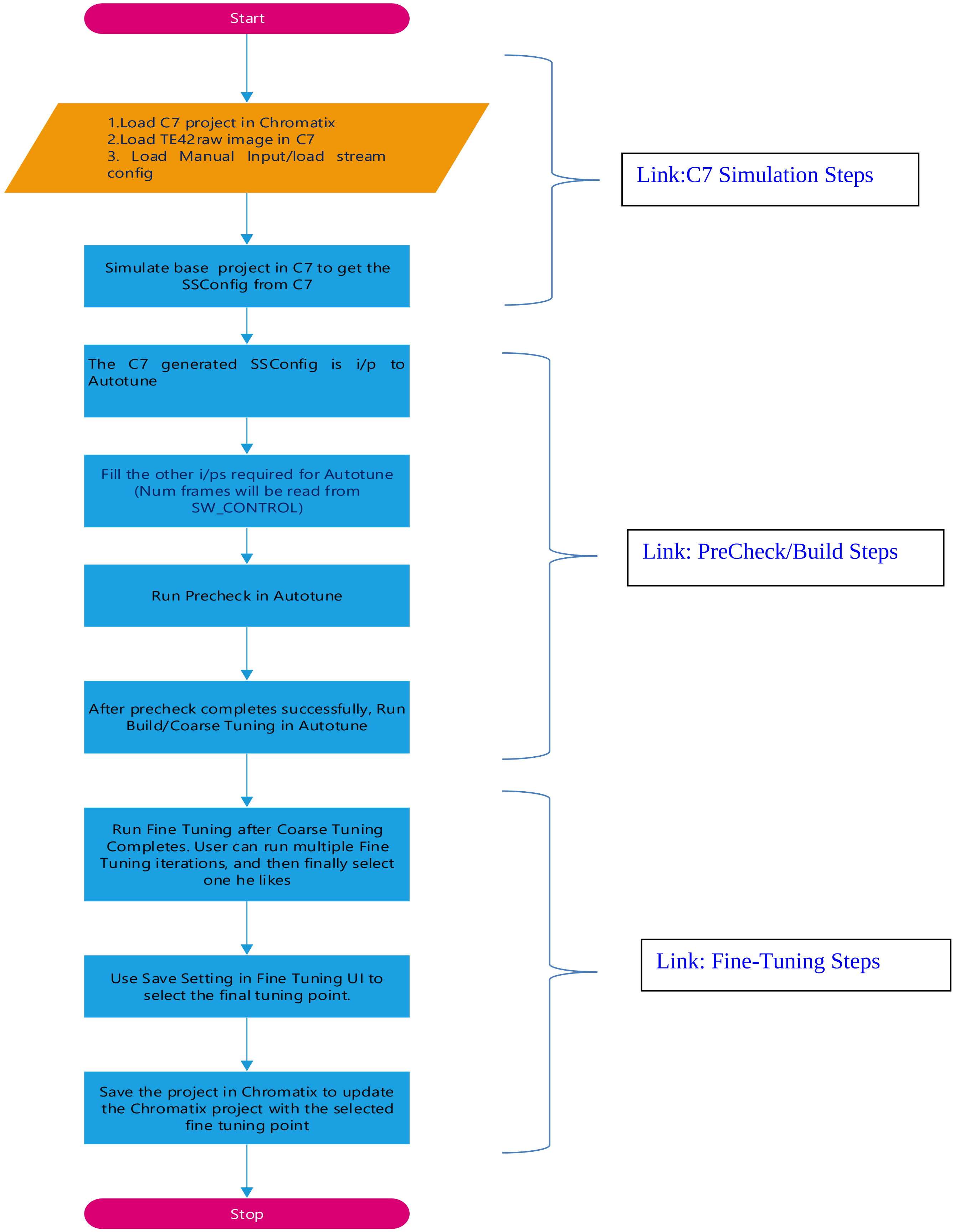

Start
1.Load C7 project in Chromatix
2.Load TE42raw image in C7
3. Load Manual Input/load stream config
Simulate base project in C7 to get the SSConfig from C7
The C7 generated SSConfig is i/p to Autotune
Fill the other i/ps required for Autotune (Num frames will be read from SW_CONTROL)
Run Precheck in Autotune
After precheck completes successfully, Run Build/Coarse Tuning in Autotune
Run Fine Tuning after Coarse Tuning Completes. User can run multiple Fine Tuning iterations, and then finally select one he likes
Use Save Setting in Fine Tuning UI to select the final tuning point.
Save the project in Chromatix to update the Chromatix project with the selected fine tuning point
Stop
Link:C7 Simulation Steps
Link: PreCheck/Build Steps
Link: Fine-Tuning Steps

Figure 3-1. Autotune Flow

3.3 Steps to Run Autotune – C7 Sim/Precheck/Build

Please follow the steps to run Autotune:

3.3.1 C7 Simulation

1. Load the base Kamorta C7 project in Chromatix, and select the correct scenario that needs to be tuned. The correct scenario needs to be selected in Chromatix as shown below for MFNR use-case. For Snapshot and Video use case select the Snapshot and Video Scenarios respectively.

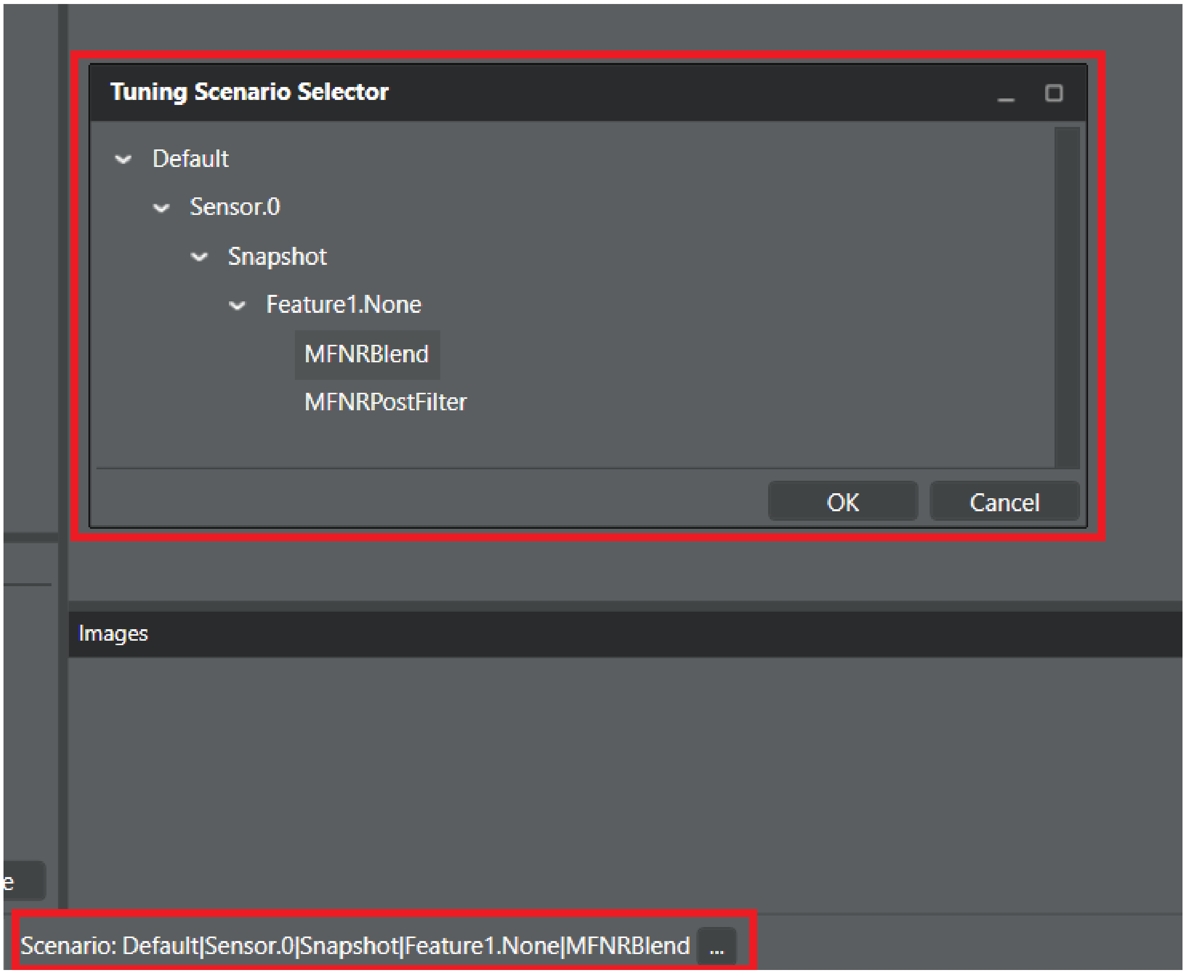


Figure 3-2 Selecting the Scenario

2. Under the Simulation tab, use the “Load Image” button to load the captured TE42 image into Chromatix.

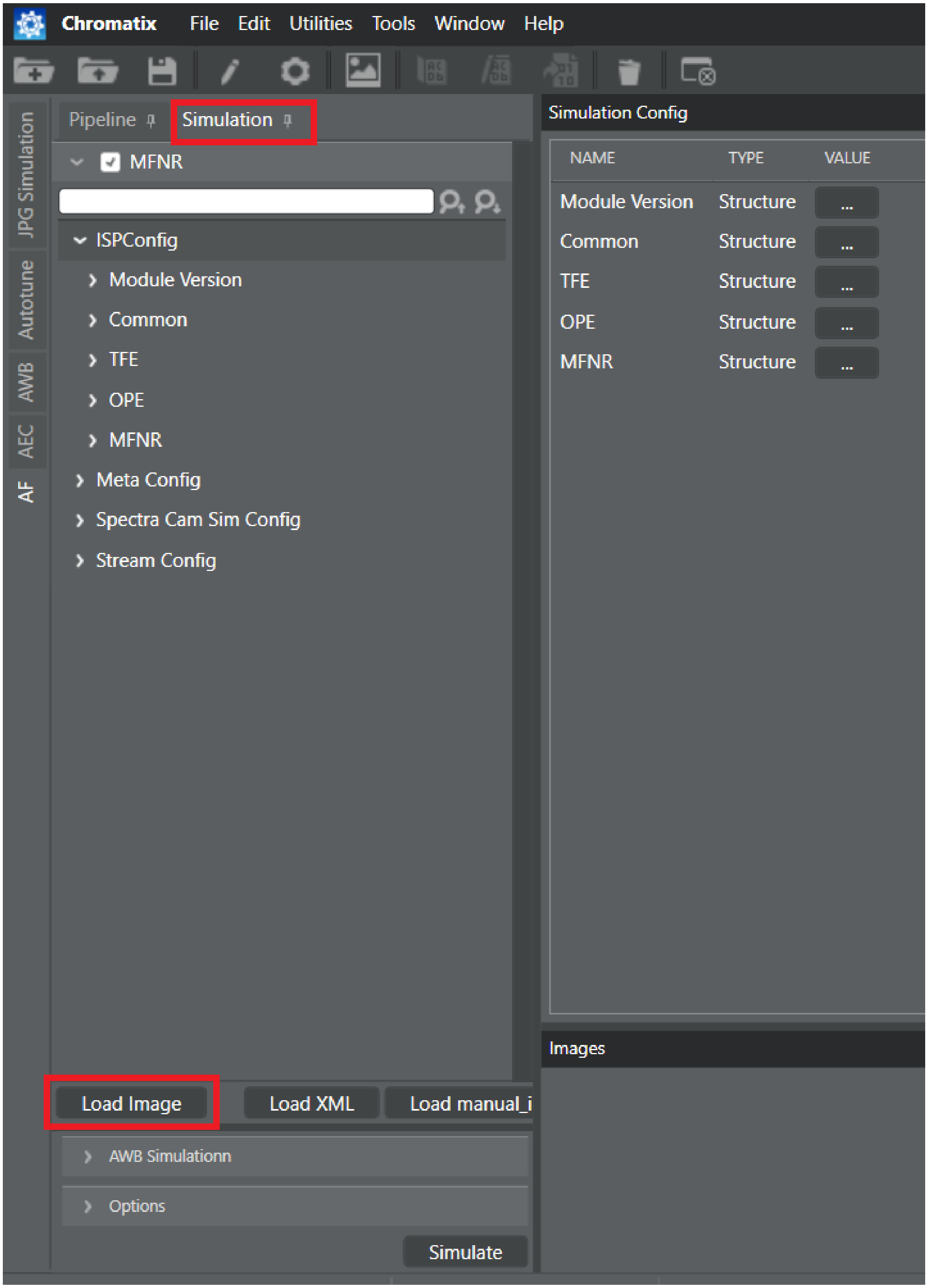


Figure 3-3 Load Image in Chromatix

Once the image is loaded, a message is displayed in the Status window of Chromatix confirming the successful loading of the image

3. After loading the image, upload the meta config file. The meta config should correspond to the region for which we are tuning currently. Select the Meta Config and then use the “Load XML” button to load the meta file.

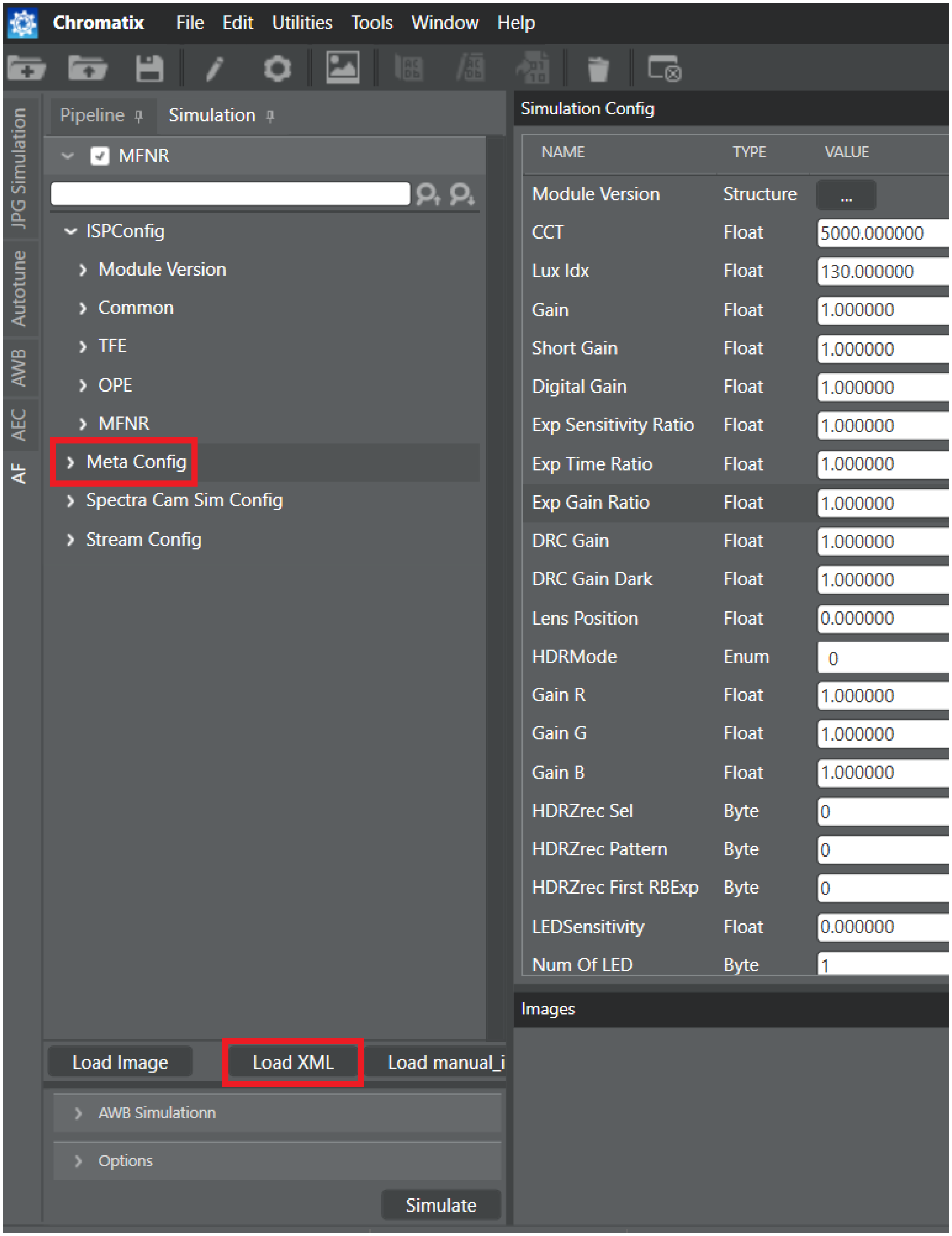


Figure 3-4 Load Meta Config in Chromatix

After loading the meta config, check if the correct values are uploaded in the meta config fields.

4. If a Manual_input file is available, select the “Spectra Cam Sim Config” and then use the “Load manual_input” button to load the manual input file. This file contains additional metadata for the input raw image. This is an optional file and may not be available with every raw file – in that case this step can be skipped.

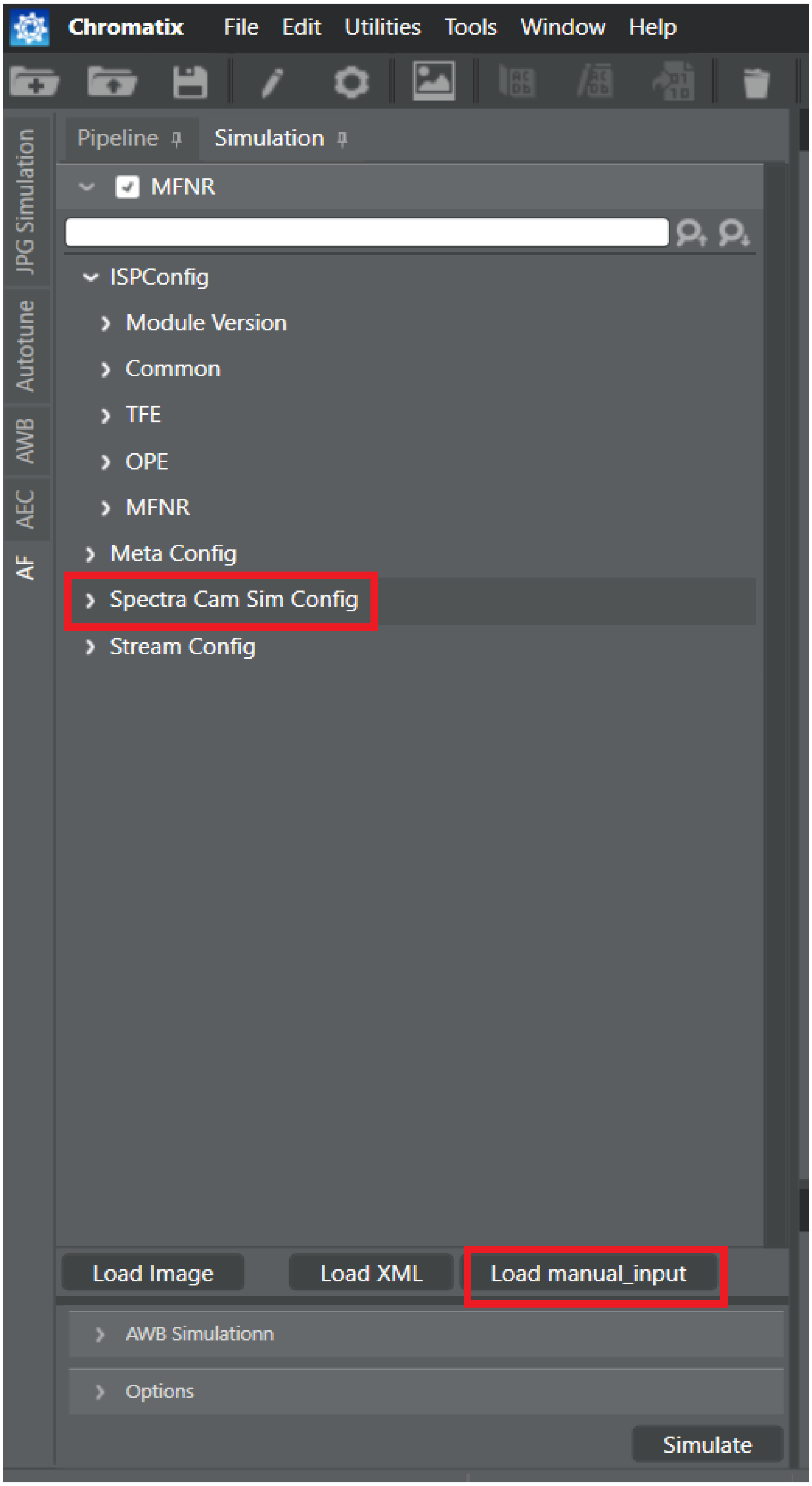


Figure 3-5 Load Manual Input in Chromatix

5. For Video Usecase, the RunPipeline in Spectracamsim config should be set OPE. This is required by the Autotune for the Video Usecase only.

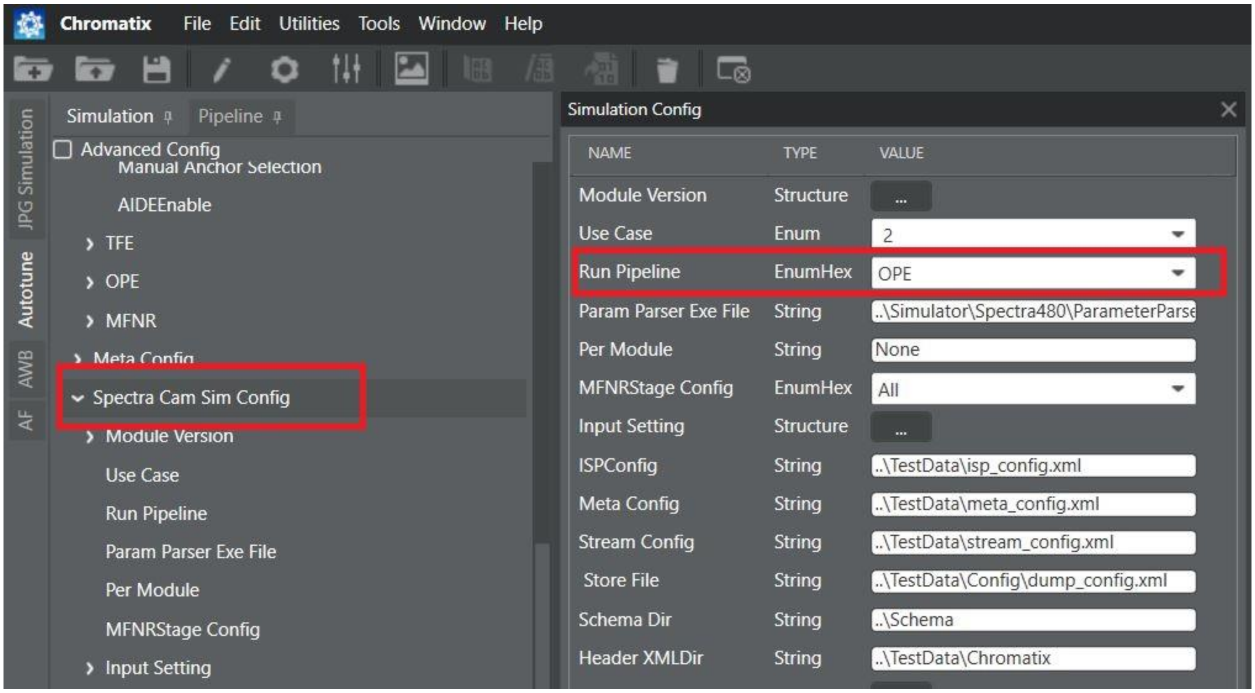


Figure 3-6 RunPipeline in Video Usecase

6. Click on the “Stream Config” and check the fields of the stream config. The format of the raw image depends on the tap-point of the captured TE42 image. For Kamorta, if the input image is 12-bit MIPI Packed, then the MIPIPaddingLength is 48. If the input image is 10-bit MIPI packed, then the MIPIPaddingLength is 80. For Kamorta(Spectra340) if the captures are Ideal Raw captures then the LSC and Demux_blklevel in TFE must be disabled. If the captures are RDI then the LSC and the Demux_blklevel must be enabled in the TFE.

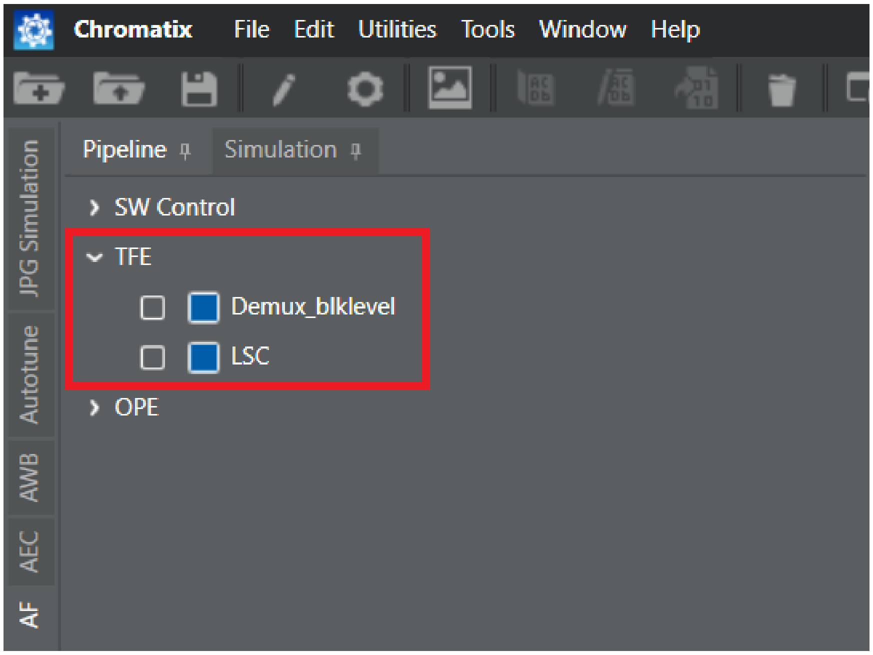


Figure 3-7 TFE in MFNR

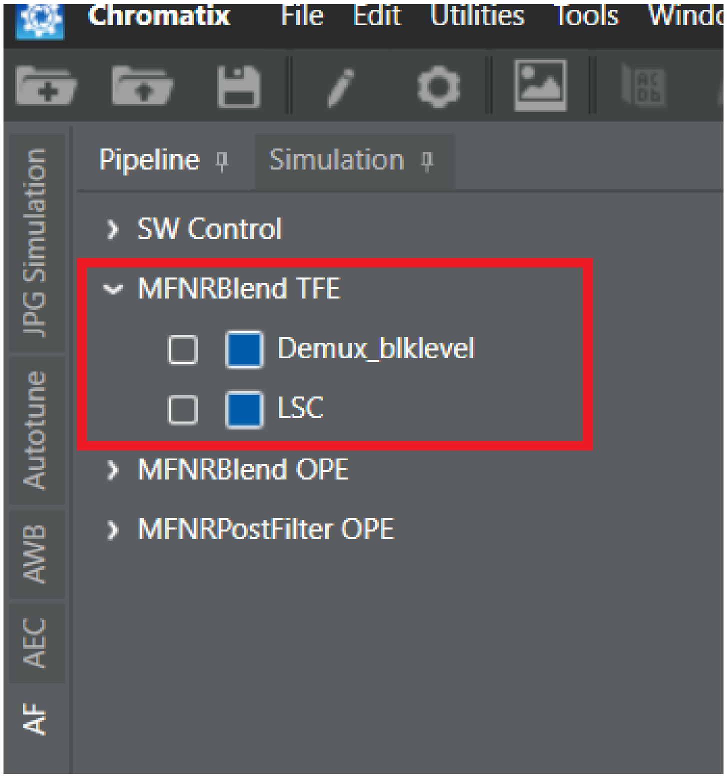


Figure 3-8 TFE in Snapshot

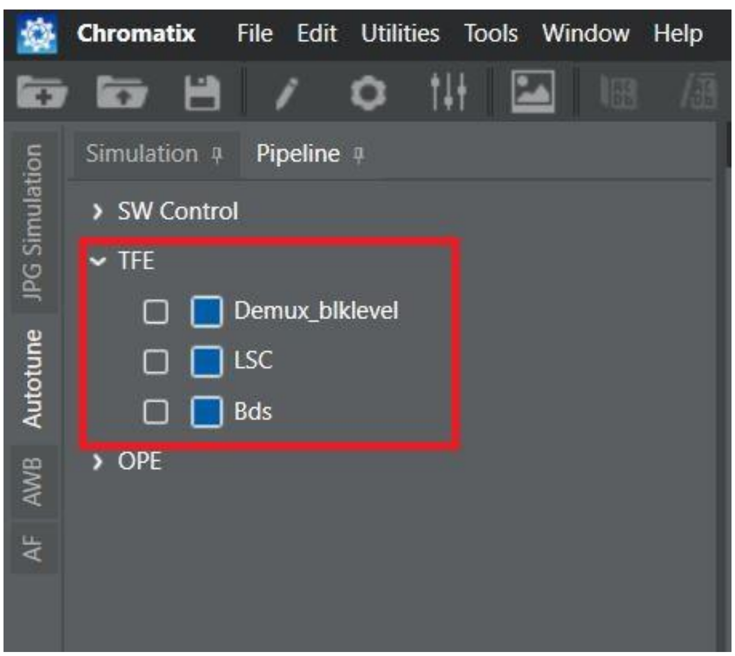


Figure 3-9 TFE in Video

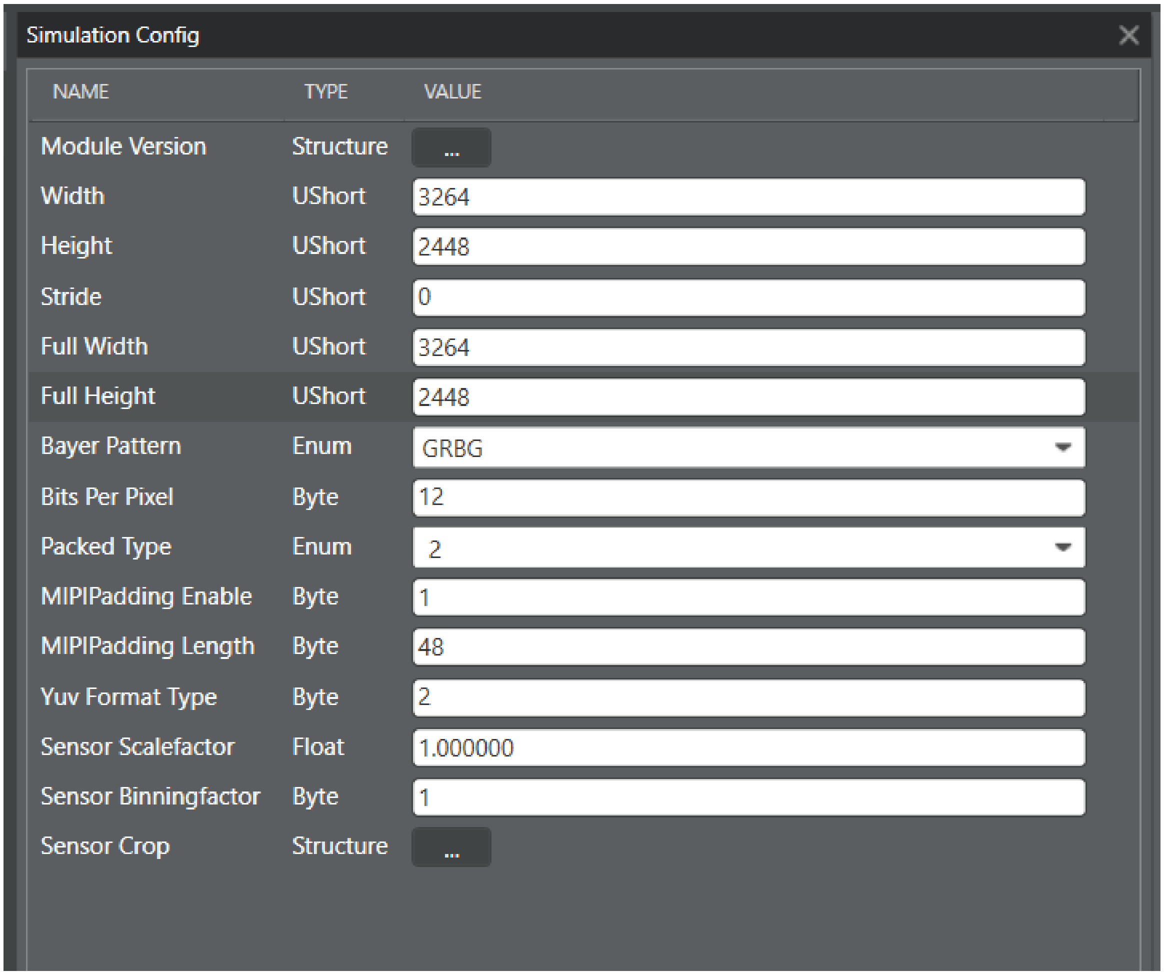


Figure 3-10 Stream Config in Chromatix

> **NOTE:** It is the user's responsibility to enter the Bits per pixel and the MIPIPaddingLength correctly in the Stream Config fields.
>
> For Kamorta(Spectra540) if the captures are Ideal Raw captures then the LSC and Demux_blklevel in TFE must be disabled in Chromatix. If the captures are RDI then the LSC and the Demux_blklevel must be enabled in the TFE in Chromatix.

7. For Video usecase, ensure that while generating the ISPConfig, correct number of frames (TotalFrameNumber) is provided. Recommended value for Autotune is minimum of 10 frames. Please note that the TotalFrameNumber is set correctly in the ISPConfig→Common→TopConfig→TotalFrameNumber field in Chromatix.

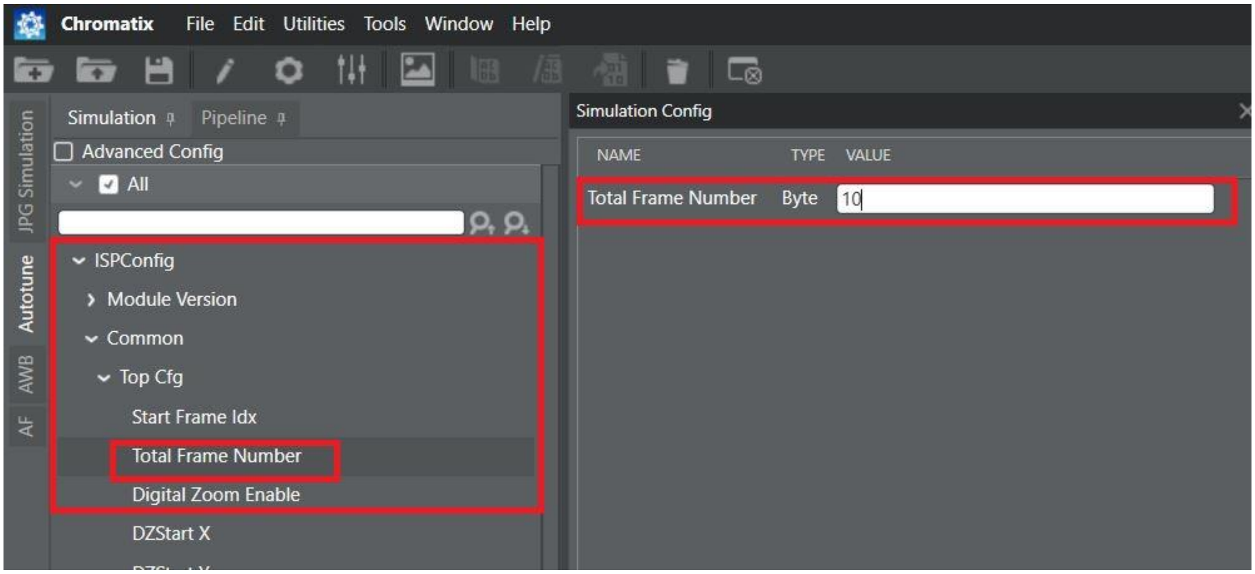


Figure 3-11 Num Frames in Video Usecase

8. Now click the “Simulate” button in Chromatix. This will do the simulation of the base C7 project for the captured TE42 image. On successful completion, the output image will be displayed in Chromatix. It’s important to check this simulation is ok since if there is an issue with this simulation, we need to resolve this before proceeding with Autotune.

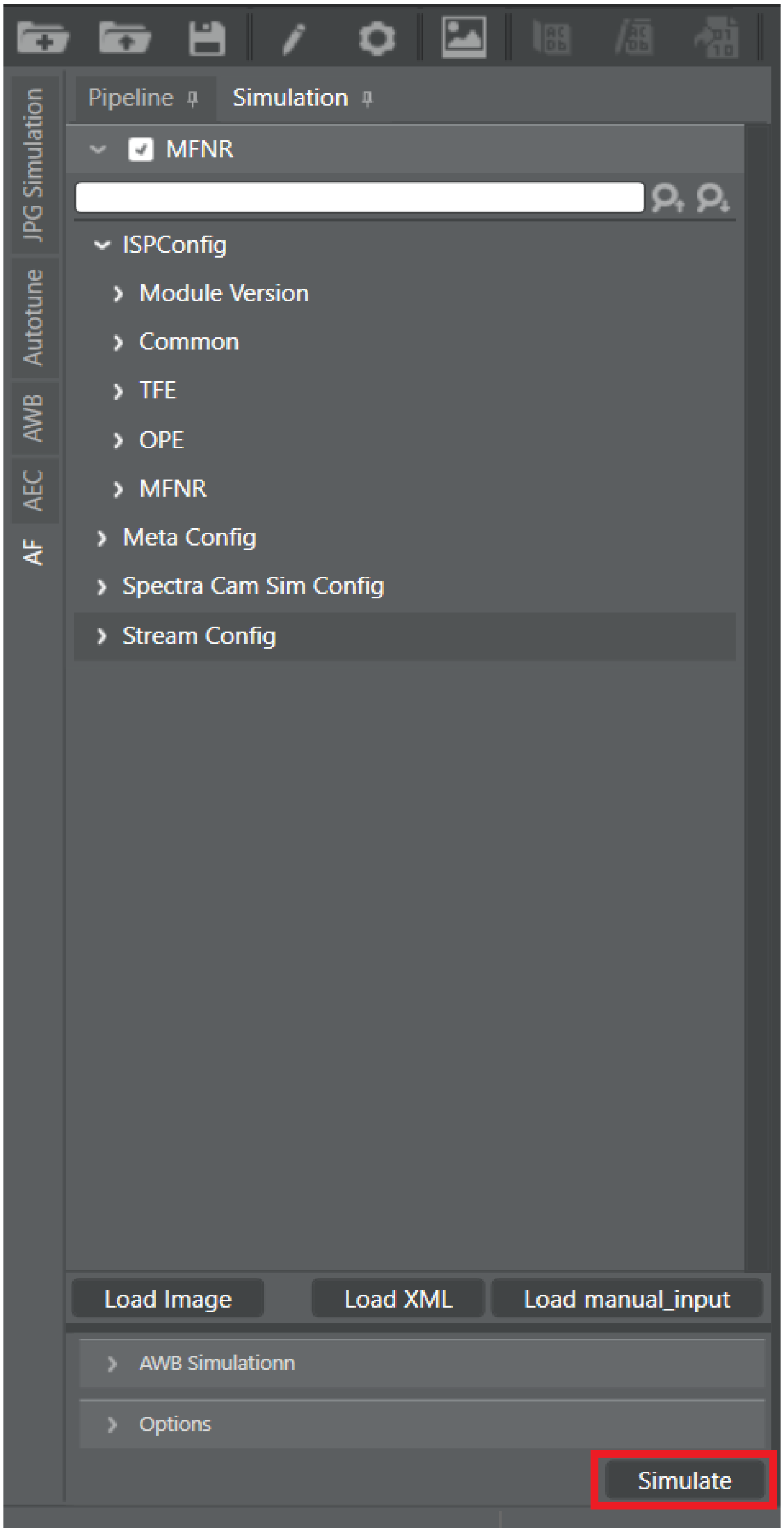


Figure 3-12 Simulate in Chromatix

9. De-Ghosting strength parameter in sw_mfnr : This parameter needs to be manually tuned before starting build database/coarse tuning in autotune. The parameter should be set such that only local motion regions are identified as in motion. Very high values of de-ghosting

strength would lead to false motion identified throughout the frame and degrade autotune performance. Very low values can lead to ghosting in local motion areas and degrade IQ in these areas.

> **NOTE:** Once the base project is simulated successfully, only then we can move to running Autotune. If there is any error in the base project simulation, we will first need to resolve this issue before running Autotune.

3.3.2 Autotune Coarse Tuning - PreCheck/Build

1. Enter the inputs to Autotune

   - Spectracamsim : This is the path to the spectracamsim config file that is generated by Chromatix in the steps 1-7 above. This is found in the Simulation folder of the C7 project. The spectracamsim config file will have the paths to the input raw images, meta config and stream config files. Please check this file to confirm if all paths are correct and the respective files are available at those locations.
   - Patch Marking: This is the path to the patch marking file. The jpeg output file which is generated needs to be patch marked as per Autotune requirements. The steps to patch mark the jpeg image is explained in detail in chapter on Patch Marking. **Note:** For Video usecase, patch marking should be done on the mctf_output_FULL.JPG of the last frame.
   - Output Path: Path to the output directory.
   - Lux : The lighting condition for which the user wants to tune. The user can select one of Bright Light, Normal Light or Mid-Light, Low Light, Ultra-Low Light conditions. The TE42 chart must be captured in the same lighting condition for which tuning is being done.
   - Delta Tuning Enable: Check this if the user thinks the base C7 project is well tuned already for noise/texture and wants Autotune to evaluate the current settings also as one of the possible tuning points.
   - ChromaEn : Autotune supports Noise/Texture tuning for Chroma with this release.
   - **NOTE : Currently, if Chroma Tuning is Enabled in the Coarse Tuning step, Fine Tuning may not give good IQ results. This will be supported in the next release of Autotune**
   - Database: The user can choose to Build a New Database when running the Autotune, or Use Existing Database to do the coarse tuning. If the user already has generated the sensor database for a particular sensor/lighting condition he could use this existing database and reduce the Autotune run time.
   - New DB Id: This is id for the current run. Also the name of the new database will include this id for easy identification.
   - Existing DB: The already generated sensor database.bin if the user selects “Use Existing Database”

- Ref Image: The reference/target jpeg file which the Autotune should use as target image to be achieved. For Video usecase, a target/reference video is not supported. It is recommended to provide the target metrics in XML format (as mentioned below).
- Ref Patch Mark : The patch marked file for the reference jpeg. The method to patch mark the reference is exactly the same as the patch marking of the simulated jpeg and is explained in detailed in chapter. This is not needed in case of Video as Target xml containing reference metrics needs to be provided directly.
- Target xml (recommended for Video usecase): Instead of the Ref Image and Ref Patch Marking file, the user can directly use a Target xml. The Target xml contains the metrics that the Autotune will try to achieve. The guidelines on how to choose the target is explained in chapter

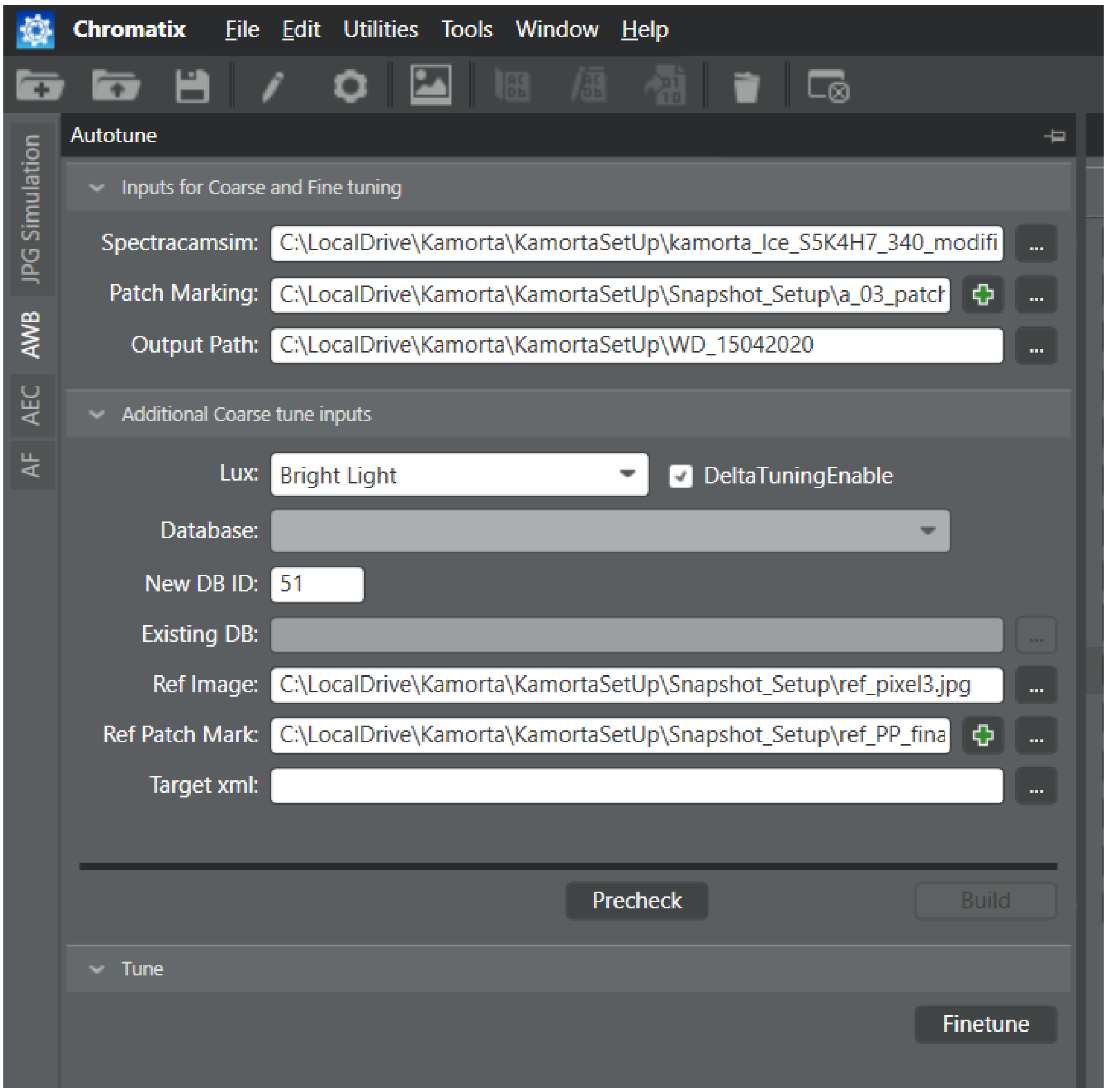


Figure 3-13 Inputs for Autotune for PreCheck and Build

2. Once the user fills in the inputs, the Precheck button in the GUI will be enabled. The user needs to click the Precheck, and then wait for Precheck to get completed. This is the step that needs to be done before running the Coarse Tuning. In Precheck, the Autotune does the checks whether the files all exist in the given path. It will also do a base simulation to check if the simulation using the input spectracamsim works fine.

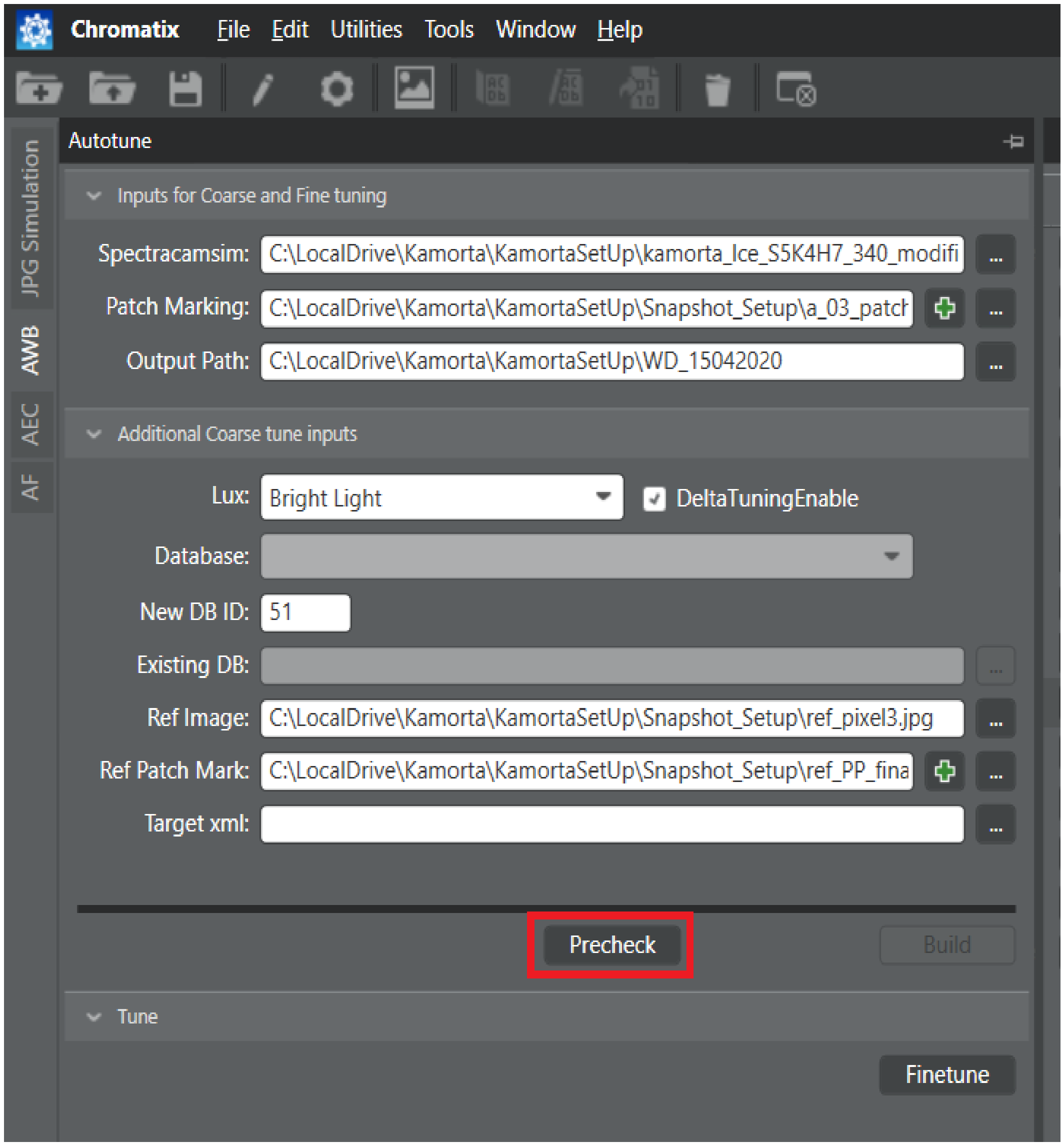


Figure 3-14 Running PreCheck in Autotune

3. Precheck will take approximately 20 mins to 45 mins to run based on the machine configuration. Once Precheck is completed, a message is displayed in the status log message will appear in the logger. The message specifies the path at which the precheck output can be found. The Outputs of Precheck will be created in Output directory/Precheck. Also, the simulated output jpeg after Precheck will be displayed in the Chromatix window as shown below. For Video usecase, Precheck displays the JPG of the last frame only.

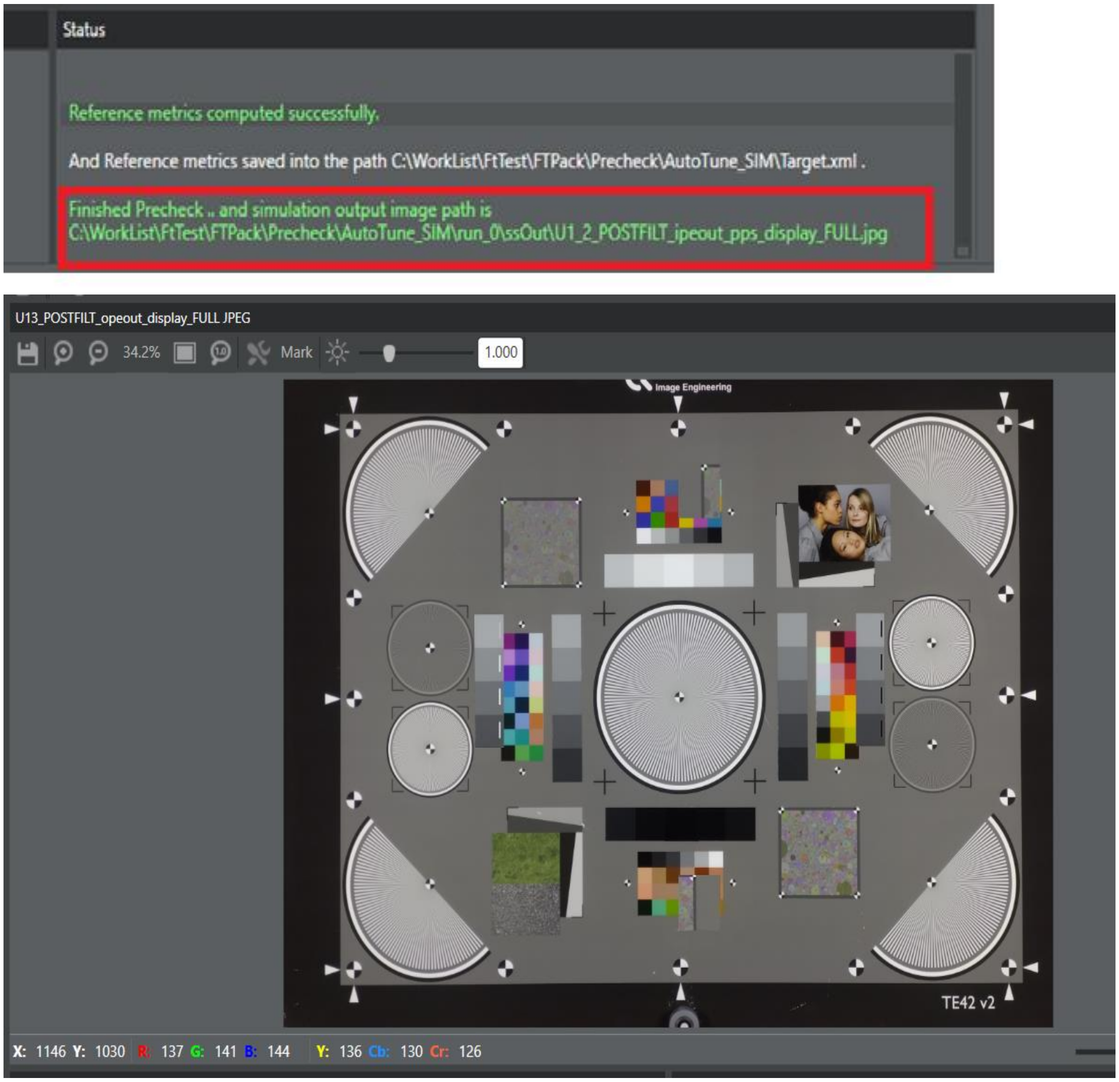


Figure 3-15 Completion of PreCheck

4. Once Precheck is completed, ONLY then "Build "needs to be clicked. This starts the Coarse Tuning. This step will take approximately 8 to 12 hrs (for Kamorta single region tuning) depending on the machine configuration and resolution of the images to be tuned. The outputs of Coarse Tuning will be created at: Output directory/ SmartU2_xx/ Coarse Tuning, where xx is the DB ID in this case.
5. IQ Analysis of Coarse Tuned outputs (recommended before FineTuning exercise)–

   a. **Snapshot and MFNR:**

      i There will be 4 JPGs in the Coarse Tuning folder (Setting_0.jpg, Setting_1.jpg, Setting_2.jpg, Setting_3.jpg) corresponding to 4 different trade-off points (coarse-tuned settings) between noise/texture/sharpness.

      ii The user can select any one of these 4 settings based on his preference.

      iii The C7 project with the tuned parameters for each of the 4 tuned settings is also present in the Coarse Tuning folder.

b. **Video:**

i In the coarse tuning folder, each of the 4 coarse tuned settings will have 10 YUV output frames each (Setting_0_fr0.yuv, Setting_0_fr1.yuv,…. Setting_0_fr9.yuv for Setting#0, Setting_1_fr0.yuv, Setting_1_fr1.yuv,…. Setting_1_fr9.yuv for Setting#1 and so on till Setting#3).

ii To analyze individual output video YUV frames, we recommend using an ImageViewer tool that supports YUV format.

iii For video analysis of a coarse tuned setting, it is suggested to re-simulate large number of frames using the corresponding setting's c7 project available in the Coarse Tuning directory.

iv Concatenate the individual frames into a single YUV video. Once concatenated (and encoded appropriately), the video obtained from the YUV frames can be analyzed using open source softwares like MPlayer/FFmpeg.

v

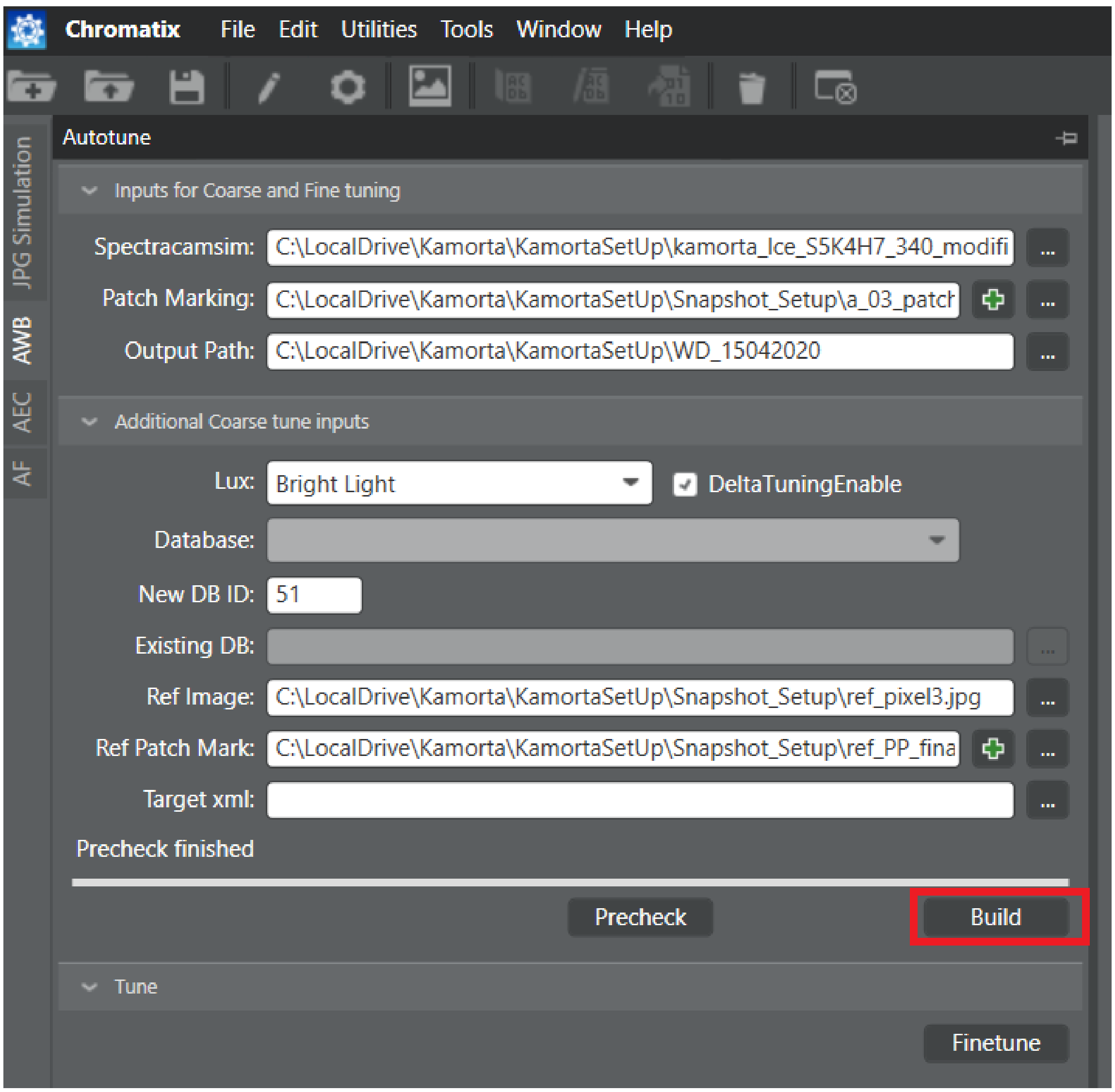


Figure 3-16 Build

6. Once the Coarse Tuning is completed, the user can click on the FineTune button to start Fine Tuning. The steps for Fine Tuning are detailed in the next section

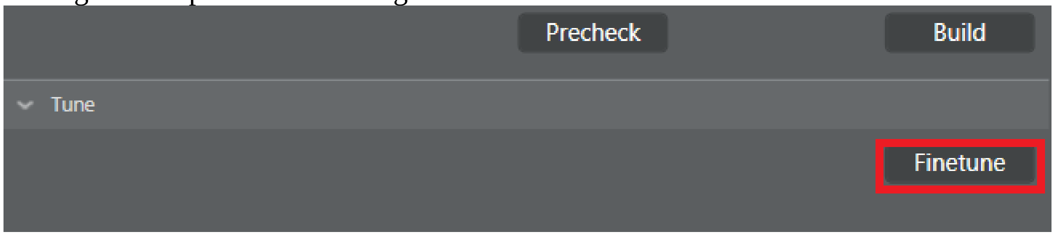


Figure 3-17 Finetune Button

3.4 Steps to Run Autotune – Fine Tune

For a particular sensor-ISP combination and lux condition (trigger region), AutoTune provides a fine-tuning framework to the user after completion of coarse tuning (using the AutoTune U2 tool) for that particular case. Using this framework, user can further access tuned settings based on his/her preferences and desired image quality enhancements.

NOTE : Currently, if Chroma Tuning is Enabled in the Coarse Tuning step, Fine Tuning may not give good IQ results. This will be supported in the next release of Autotune

The steps for fine tuning using the Autotune tool are below:

1. Open Chromatix and load the base project (preferably, this project should be same as that used to run coarse tuning for the desired case).
2. Select the desired scenario folder for tuning and generate the spectracamsim_config.xml by providing all necessary config files and other details. ***Note:*** *This step can be skipped if you already have a valid copy of the spectracamsim_config.xml (with correct paths) that was created during the coarse tuning exercise.*
3. Select the AutoTune tab (in the left) and fill the set of common inputs (Inputs for coarse and fine tuning):
    a. Spectracamsim: Provide path to a valid spectracamsim_config.xml (It should contain all valid paths).
    b. Patch Marking: The patch property XML (this must be same as that used for coarse tuning).
    c. Output path: Provide a work directory path.

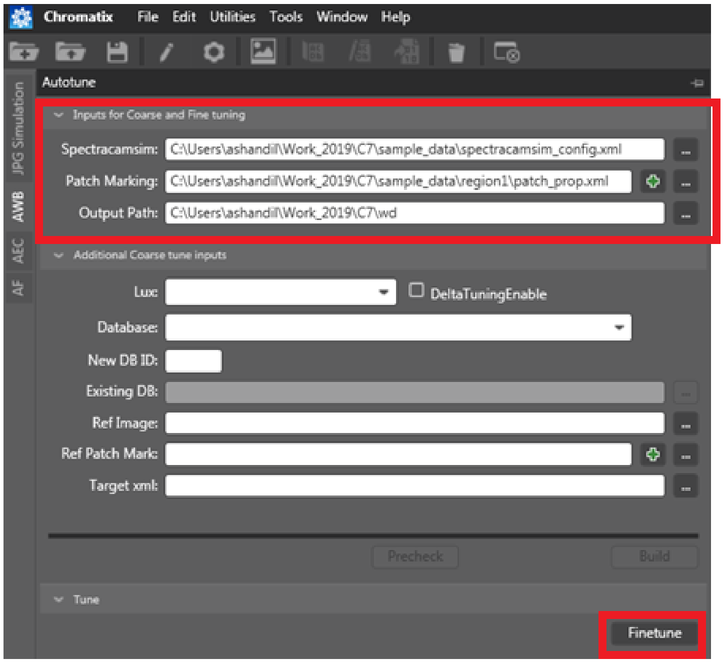


Figure 3-18 Inputs for Finetune

4. Click on the “FineTune” button. This will pop up a new GUI.
5. Browse the path to the U2 output folder (e.g. SmartU2_051) which you obtained during coarse tuning for this case and click “Load”. *(Note: This folder should contain the “CoarseTuning” and “db” folders)* → This will load the coarse tuning output settings in the GUI.

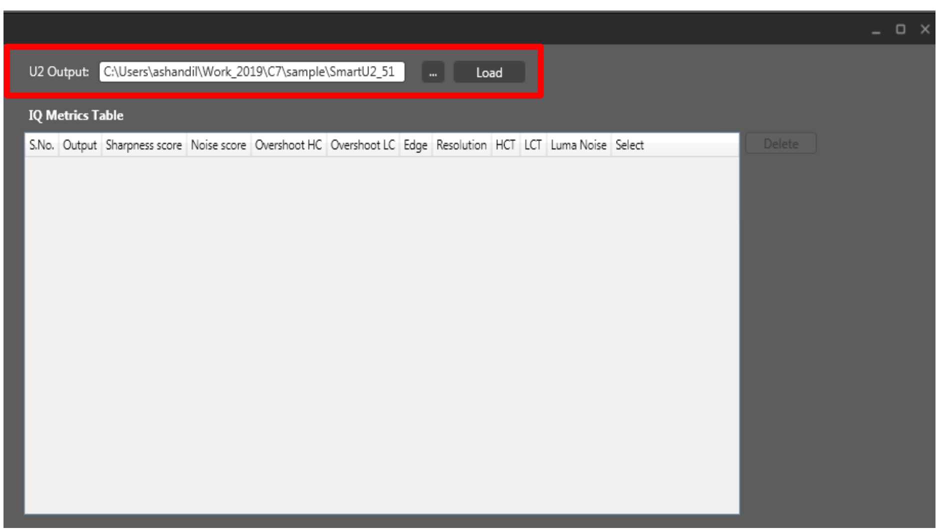


Figure 3-19 Load Coarse Tune Folder

6. **Select for FineTune:** Select the options/user directions for fine tuning on the bottom left of the GUI:
   a. *Output:* Based on metrics/scores of the coarse tuned settings in the loaded table and visual inspection done on the corresponding images (already dumped in CoarseTuning folder during U2 run), choose the setting which you want to improve from the Output selection dropdown list.

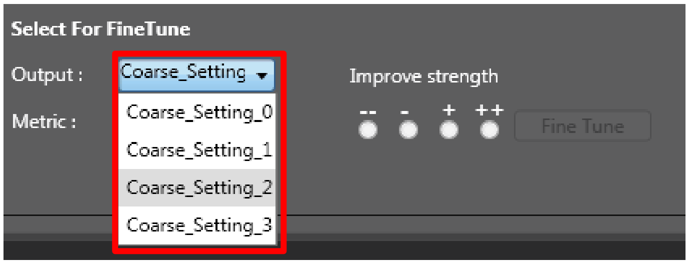


Figure 3-20 Selection of the Coarse Tuning Output

   b. *Metric:* Based on the type of enhancement required on the selected setting, choose an image quality metric out of Texture, Resolution, Edge Sharpness, Luma Noise, Overshoot.
   Note: The metrics displayed for Video usecase will correspond to IQ on the last frame.

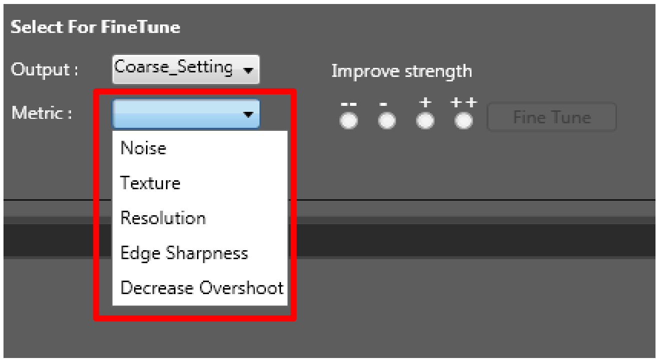


Figure 3-21 Selection the Metric for Tuning

   c. *Improve Strength:* Direct the strength (amount) of change desired on the selected metric. For e.g. If you want to reduce noise by a small amount, choose "- ". If you want to reduce noise on the same setting relatively more, then choose "- - ". Similarly, to increase a metric (say Texture), choose "+" or "++" based on the amount of change desired. **Note:** Improve strength option is not applicable if user selects "Decreasing Overshoot".

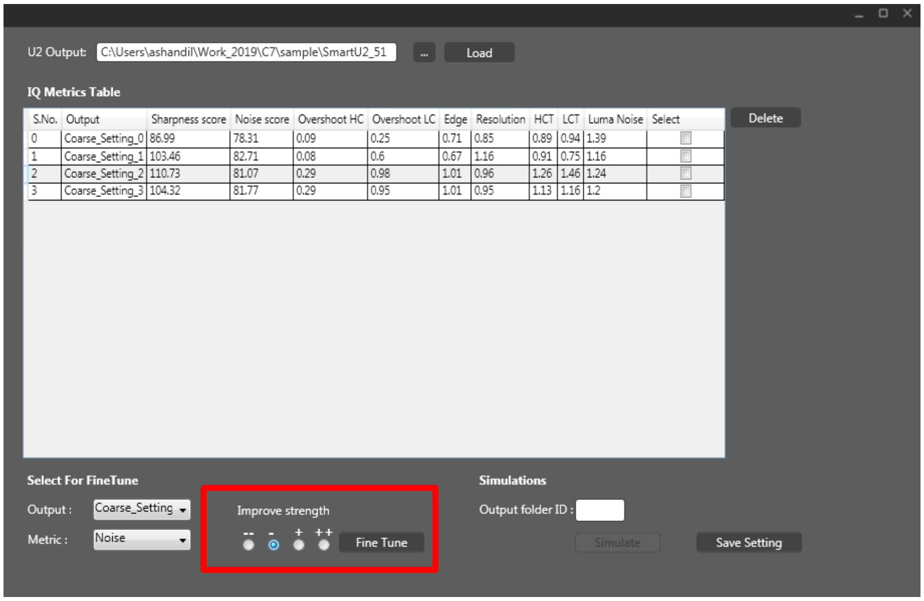


Figure 3-22 Select Metric Strength for Tuning

> **Note**: It is possible that some of the 4 outputs from Coarse Tuning are identical. In such a case the Fine Tune GUI will only display the unique points out of the 4 Coarse Tuning Points.

d. Click "Fine Tune" → This updates the IQ Metrics table with new fine-tuned settings based on the above selections.

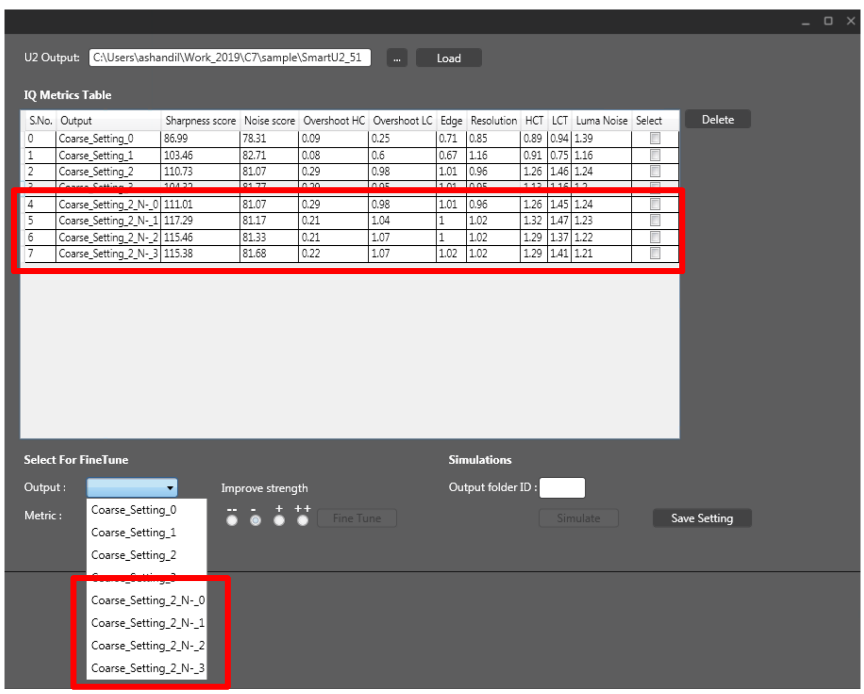
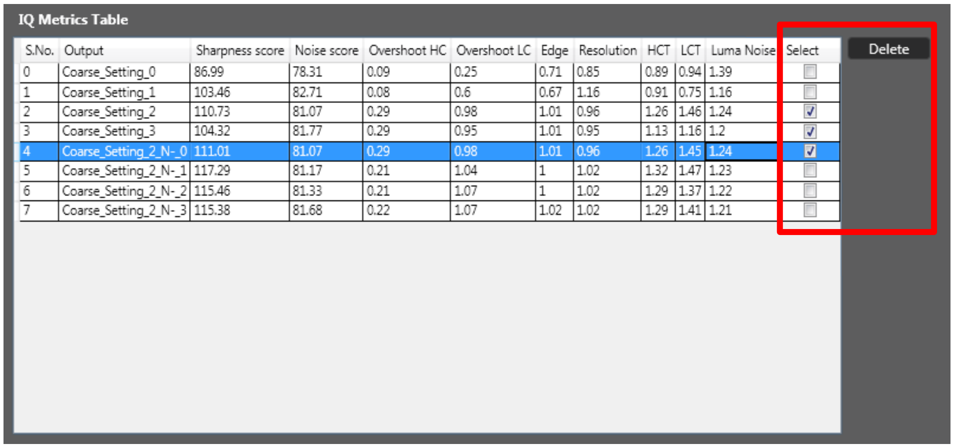


Figure 3-23 Table updated with new Fine-tuned points

e. Repeat steps (a) to (d) by choosing any new or previously selected setting and directing required IQ metric improvements till you are satisfied with the output. E.g. After reducing Noise on Coarse_Setting_2 by "-", you can now select Coarse_Setting_2_N-_2 and "Decrease Overshoot" or you can try to increase Noise/Texture on Coarse_Setting_3 instead.

7. **Naming Convention:** The name of the new fine-tuned output settings is derived by appending the first alphabet of the selected metric and the improve strength as a suffix to the name of the originally selected setting. For e.g. If selected output is Coarse_Setting_2, selected metric is Noise and selected improve strength is "-", and the GUI provides 4 new fine-tuned outputs, they will be called Coarse_Setting_2_N-_0, Coarse_Setting_2_N-_1, Coarse_Setting_2_N-_2, Coarse_Setting_2_N-_3.

8. **Delete:** You can delete any undesired setting from the table by double-clicking the checkbox under the "Select" column for that setting, followed by clicking "Delete".


IQ Metrics Table

S.No.	Output	Sharpness score	Noise score	Overshoot HC	Overshoot LC	Edge	Resolution	HCT	LCT	Luma Noise	Select
0	Coarse_Setting_0	86.99	78.31	0.09	0.25	0.71	0.85	0.89	0.94	1.39	☐
1	Coarse_Setting_1	103.46	82.71	0.08	0.6	0.67	1.16	0.91	0.75	1.16	☐
2	Coarse_Setting_2	110.73	81.07	0.29	0.98	1.01	0.96	1.26	1.46	1.24	☑
3	Coarse_Setting_3	104.32	81.77	0.29	0.95	1.01	0.95	1.13	1.16	1.2	☑
4	Coarse_Setting_2_N-_0	111.01	81.07	0.29	0.98	1.01	0.96	1.26	1.45	1.24	☑
5	Coarse_Setting_2_N-_1	117.29	81.17	0.21	1.04	1	1.02	1.32	1.47	1.23	☐
6	Coarse_Setting_2_N-_2	115.46	81.33	0.21	1.07	1	1.02	1.29	1.37	1.22	☐
7	Coarse_Setting_2_N-_3	115.38	81.68	0.22	1.07	1.02	1.02	1.29	1.41	1.21	☐

Delete

Figure 3-24 Deletion of Unwanted Tuned Settings

9. **Simulation:** After shortlisting some settings, you can obtain the corresponding simulated images and updated C7 project by selecting those settings by double-clicking the check-box under the select column, providing an "Output Folder ID" at the bottom right and clicking "Simulate". **Note:** You can select multiple settings for simulation, but it is advised to not select too many at once so that it does not take a lot of time and do shortlisting based on metrics before selecting for simulation instead of simulating all the points. The Output Folder ID is used to name the folder which will contain the images and updated C7 project for the selected settings and is required to be unique every time you simulate a set of points. Status of simulation will be displayed in the Chromatix status window.

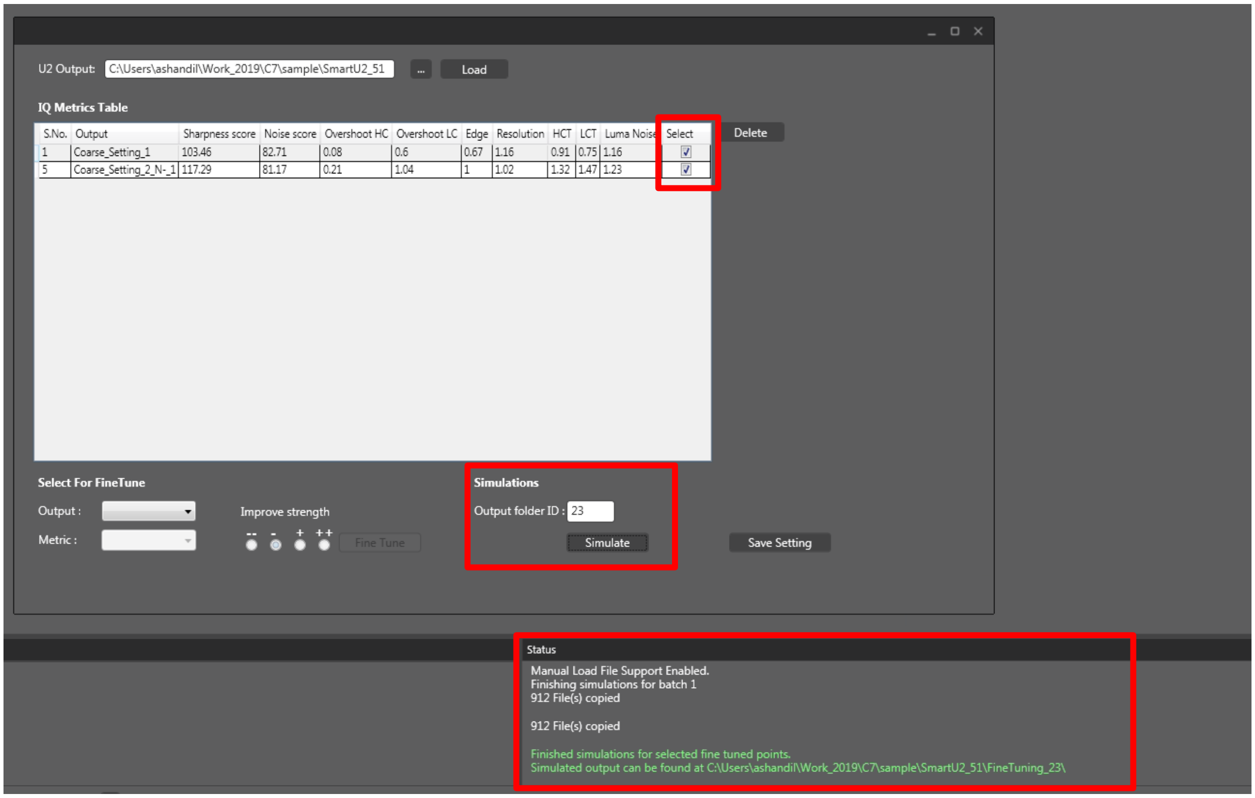


Figure 3-25 Simulation of Selected Points

10. **Save Setting :** After running simulation and analyzing the simulated outputs, the user can choose to select any one of the points as the final tuned point. To save this tuned point, the user will need to select the point by double-clicking the checkbox and then pressing the Save Setting button. **Note:** The point is not yet saved into C7 project. After clicking "Save Setting". The user will need to save the project in the Chromatix GUI to finally save the tuned parameters into C7 project.

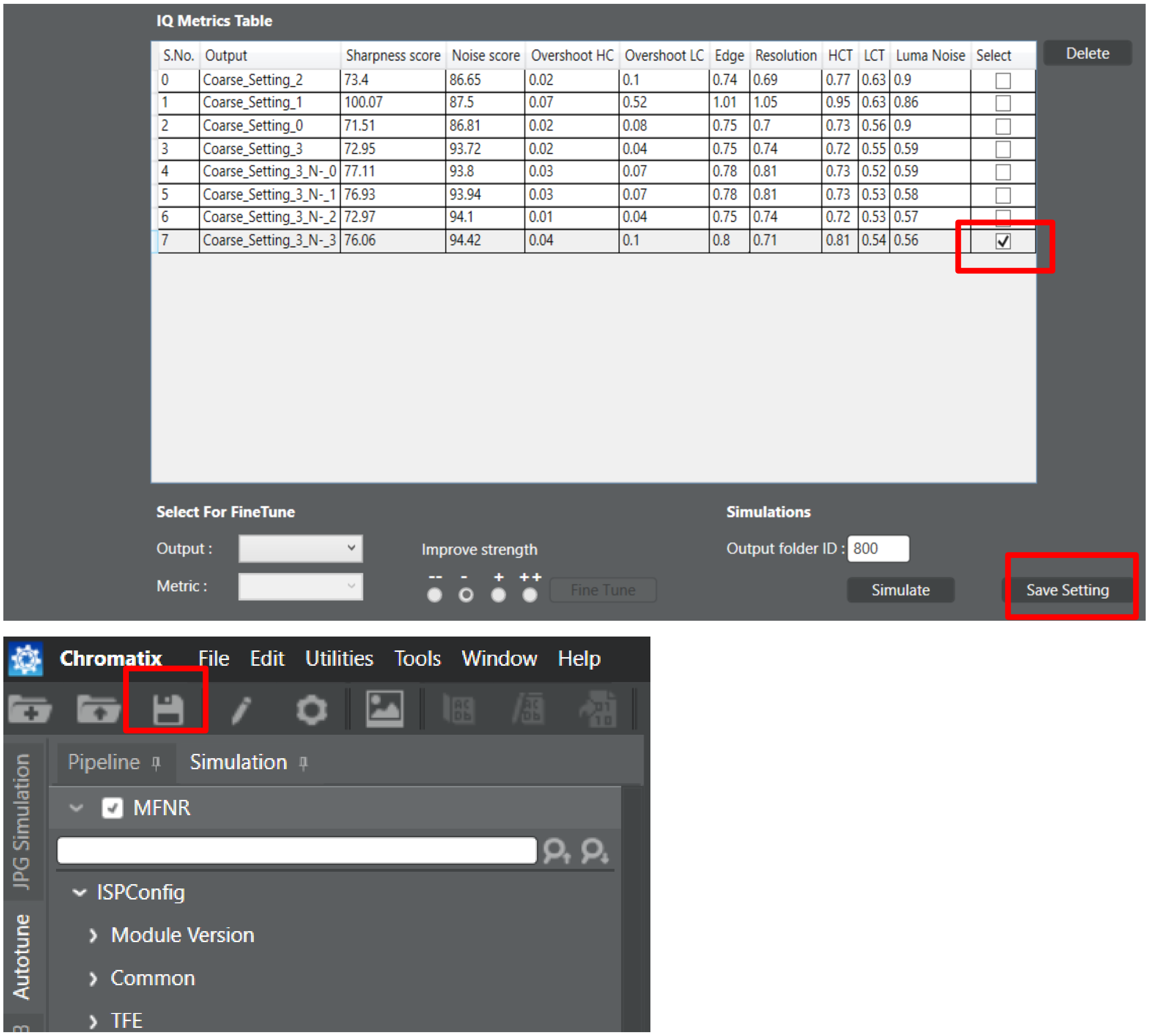


Figure 3-26 Save the Selected Finetuned Point

3.5 Metrics Scores

The recommended ranges for the metrics for Kamorta are shown in the table below for Snapshot and MFNR usecase. Please note that the metrics will vary for the different lighting conditions and sensors.

Metric	Lower bound	Upper bound	Desirable values (Bright, mid, low light)		
Edge MTF	0	1.2	>0.9	>0.75	>0.6
Resolution MTF	0	1.2	>0.9	>0.75	>0.6
LC Texture MTF	0	1.2	>0.8	>0.65	>0.5

HC Texture MTF	0	1.2	>0.9	>0.75	>0.6
Luma Noise	0	2	Closer to 0		
Chroma Noise	0	3	Closer to 0		
Chroma Saturation HC	0	26	> 20	>15	>10
Chroma Saturation LC	0	26	>10	>7	>5
Overshoot HC	0	-	<0.2		
Overshoot LC	0	-	<0.1		
VN	0	100	Higher values means lower noise.		
VS	0	120	Can be higher.		

The recommended ranges for the metrics for Kamorta for Video usecase (1920x1080) are shown in the below table.

Metric	**Lower bound**	**Upper bound**	**Desirable values (Bright, mid, low light)**		
Edge MTF	0	1.2	>0.9	>0.85	>0.8
Resolution MTF	0	1.2	>0.7	>0.7	>0.6
LC Texture MTF	0	1.2	>0.2	>0.1	>0.0
HC Texture MTF	0	1.2	>0.35	>0.2	>0.15
Luma Noise	0	2	Closer to 0		
Chroma Noise	0	3	Closer to 0		
Chroma Saturation HC	0	26	>15	>10	>5
Chroma Saturation LC	0	26	>5	>3	>1.5
Overshoot HC	0	-	<0.2		
Overshoot LC	0	-	<0.1		
VN	0	100	Higher values means lower noise.		
VS	0	120	Can be higher.		

Where

Sharpness Score - VS = 50*resmtf+ 50* hctmtf

Noise Score - VN = 20*(5 – max(meanVN-0.5, 0))

Where meanVN = sum of Luma noise and chroma noise.

3.6 Modules and Parameters Tuned by Autotune

The modules tuned by Autotune for the Snapshot and MFNR use cases in Kamorta are :

Snapshot – ABF, WNR, ASF

MFNR - SW_MFNR, WNR, ASF

Video – WNR, ASF, SWMCTF

The parameters for each of the modules that are tuned by Autotune are shown in the table below:

Module	**Parameters tuned by Autotune**	
	Luma	Chroma
ABF	noise_prsv_anchor_lo noise_prsv_anchor_hi noise_prsv_lo noise_prsv_hi edge_softness	
WNR	denoise_edge_softness_y_tab denoise_scale_y_tab denoise_weight_y_tab	denoise_edge_softness_c_tab denoise_scale_c_tab denoise_weight_c_tab
ASF	layer_1_hpf_symmetric_coeff, layer_1_gain_positive_lut, layer_1_gain_negative_lut, layer1_gain_weight_lut gain_contrast_positive_lut gain_contrast_negative_lut	
SW_MFNR	local_motion_smoothing_strength dtf_luma_strength	dtf_chroma_strength (not fully enabled)
SW_MCTF	spatial_denoising_strength (enableMC=1) [**Pre-requisite** – blend_strength and motion_detection_sensitivity are well tuned in the base project]	

Please note that Chroma parameters can be enabled through Chroma Enable flag in Autotune config file. In the current release of Autotune, the Chroma parameters are disabled.

4 Patch Marking

Autotune uses TE42 image (Figure 4-2) for tuning. Specifically, the following five regions in the TE42 image are used

1. Central resolution chart
2. High contrast and low contrast slanted edges
3. High contrast dead leaf chart
4. Low contrast dead leaf chart
5. 20 noise patches

These regions are shown in Figure 4-3

Details of how to mark these noise patches are provided in the below sections. To facilitate marking these patches, a patch marking GUI is provided as part of the Chromatix Autotune package.

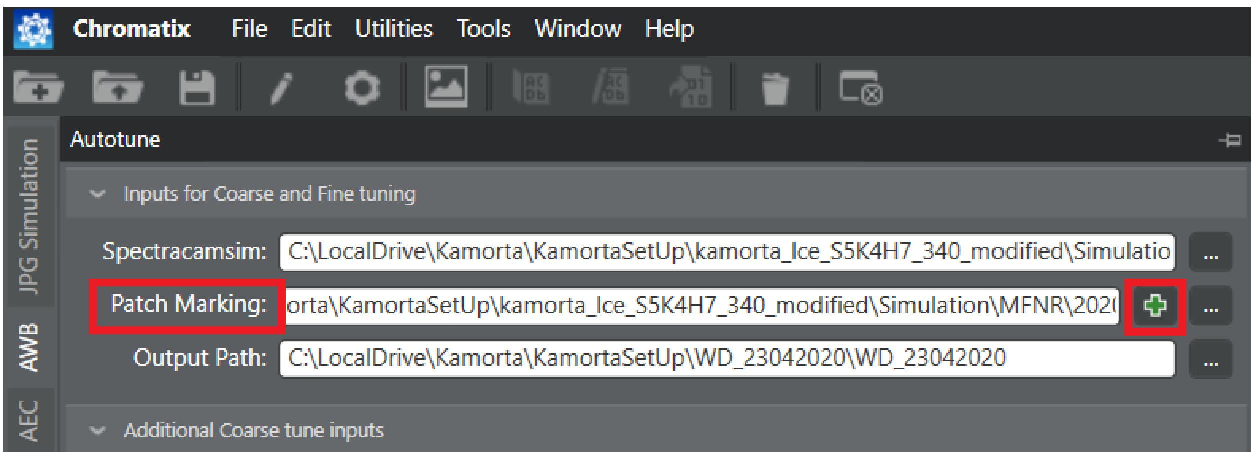


Figure 4-1 Patches used for metric computation

This is invoked by double clicking on the "+" as shown in the After the GUI opens load the TE42 image file. The supported image formats are .jpg, .bmp and .png.

This will open the TE42 image for patch marking. This window in shown in Figure 4-2. See Sections 4.2.1 to 4.2.5 for detailed instruction on how to mark the patches for each of 5 different regions.

Figure 4-2 TE42 chart which needs to be patch marked for use by Autotune. This screenshot shows the initial window when the TE42 image is loaded into the patch marking GUI

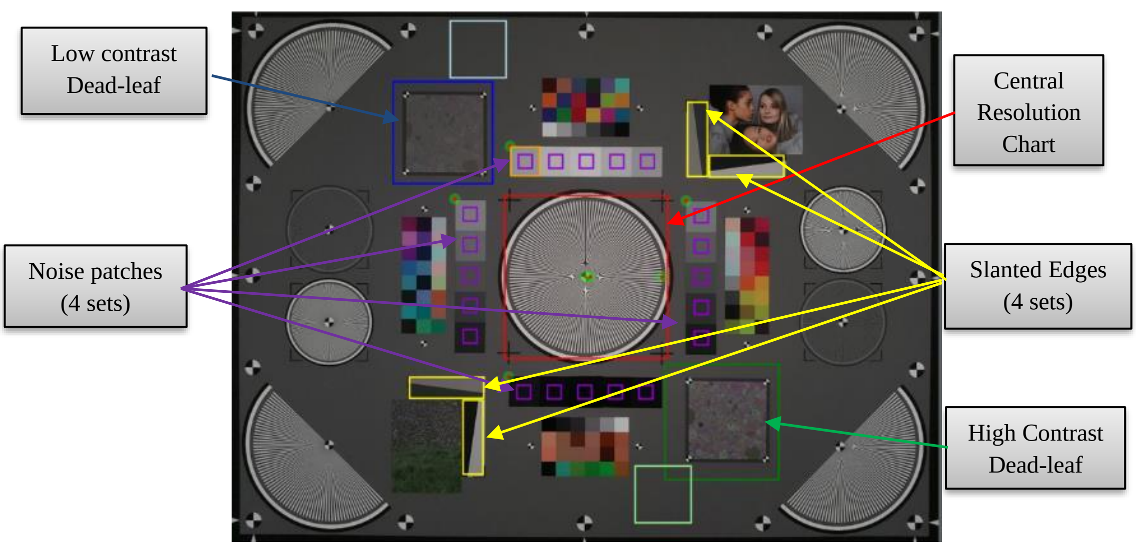


Figure 4-3 Patches used for metric computation

4.1 Automatic Detection of Patches

The tool provides an option to automatically detect the patches in the TE42 image. After this, the required patches are marked in the image. If the auto-detect throws error or some of the patches are not detected properly, it can be marked again as per the details in section 8.2 After this, clicking the Done button allows the user to save the patches into an XML file.

Post this, the refinement of the dead leaves chart needs to be carried out as per section **Error! Reference source not found.**.

4.2 Patch marking

4.2.1 Resolution chart

This consists of 2 steps, selection of the central resolution chart and ***precise*** marking of the center, inner circle and outer circle.

4.2.1.1 Select an ROI box for starchart

Select "Resolution MTF" radio button under "Metric selection" on top-left of the GUI.

Click mouse right button and select "Select an ROI box for starchart", use mouse left button to draw a rectangle around the resolution chart as shown in Figure 4-4. The entire resolution chart including the four "plus" signs should be present in the rectangle.

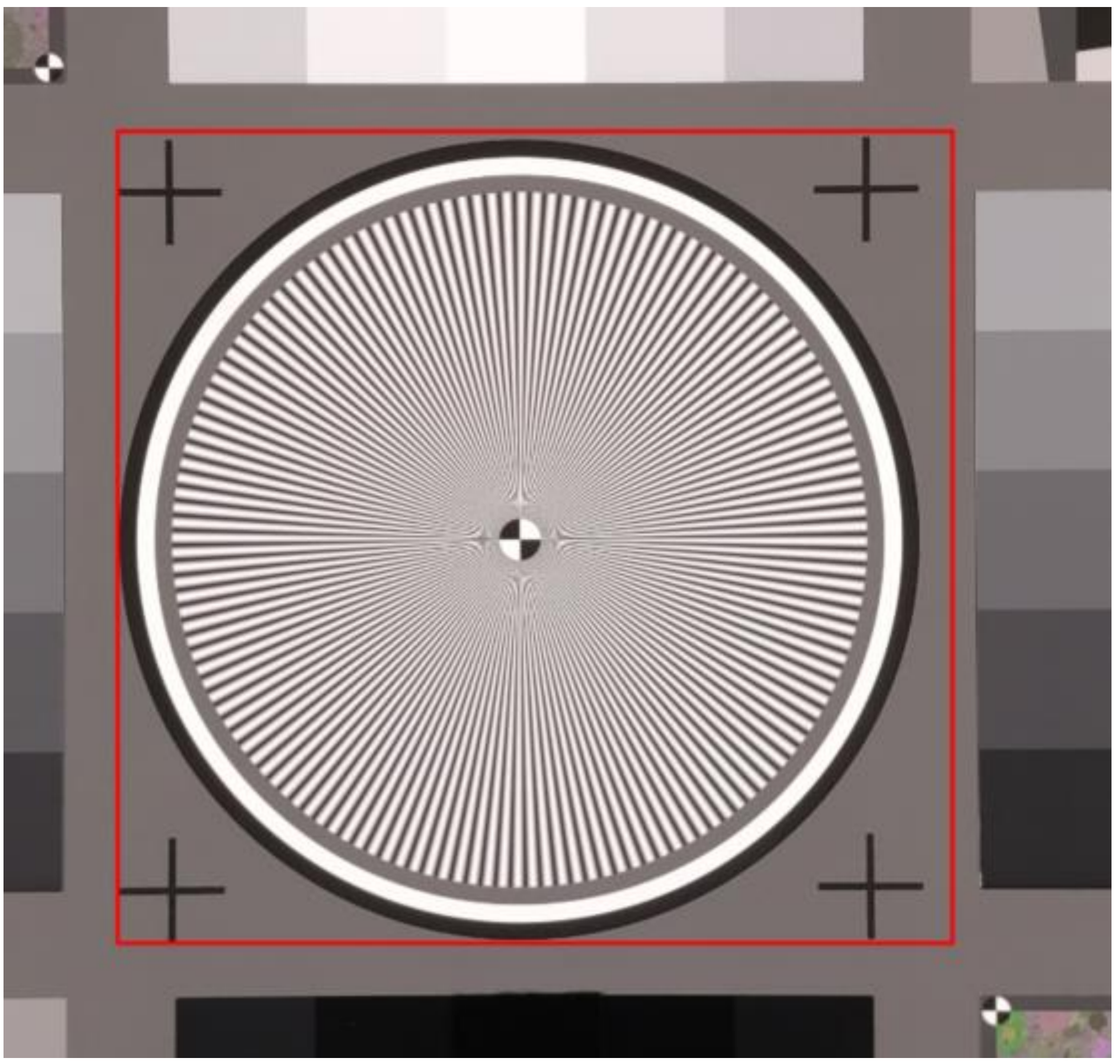


Figure 4-4 Central resolution chart.

4.2.1.2 Select center, inner radius and outer radius

Click mouse right button and select "Select center, inner radius and outer radius". These need to be marked in order. Zoom in to accurately mark these as resolution MTF computation is sensitive to accurate marking of these points.

1. Center: mark the center of resolution chart this is intersection of the black and white circle sectors at the middle of the resolution chart.
2. Inner radius: this is the radius of the inner circle
3. Outer radius: this is radius of the resolution chart.

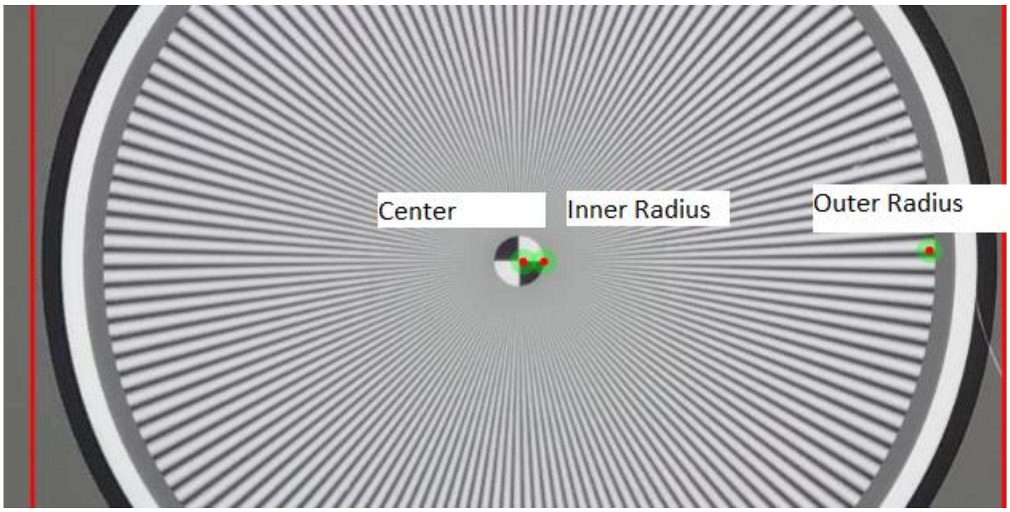


Figure 4-5 Center, inner circle and outer circle for resolution chart

4.2.2 Edge patch

Mouse right click, select "Select SlantedEdgeTRHC", mark the high contrast slanted edge in the top right portion of the TE42 chart by drawing a rectangle as shown in Figure 4-6. Make sure bottom left of the rectangle lies in the white area and top right of the rectangle lies in the black area of the slanted edge area. Select as large a rectangle as possible.

Figure 4-9 shows an incorrectly marked edge patch. The bottom left of the marked rectangle lies in black area. It should be in the white area (as shown by the green dot).

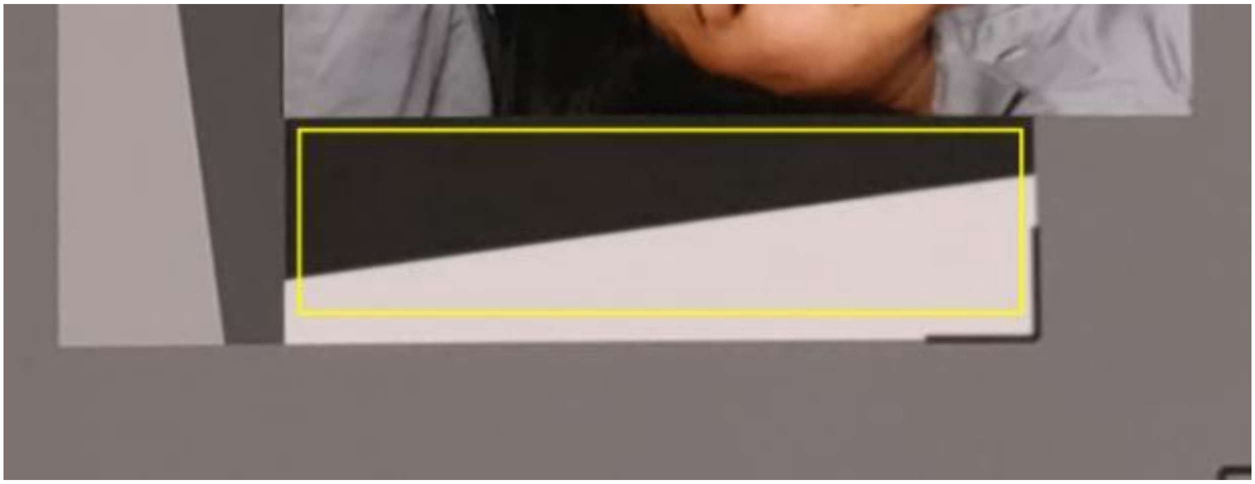


Figure 4-6 High contrast top right slanted edge patch. Bottom left of the rectangle should be in white area and top right of the rectangle should be in the black area.

Similarly select rectangle regions around low contrast edge patch in the top right region by selecting “SlandtedEdgeTRLC” (see Figure 4-8), high contrast edge patch in the bottom left region by selecting “SlantedEdgeBLHC” (see Figure 4-7) and low contrast edge patch in the bottom left region by selecting “SlantedEdgeBLLC” (see Figure 4-8).

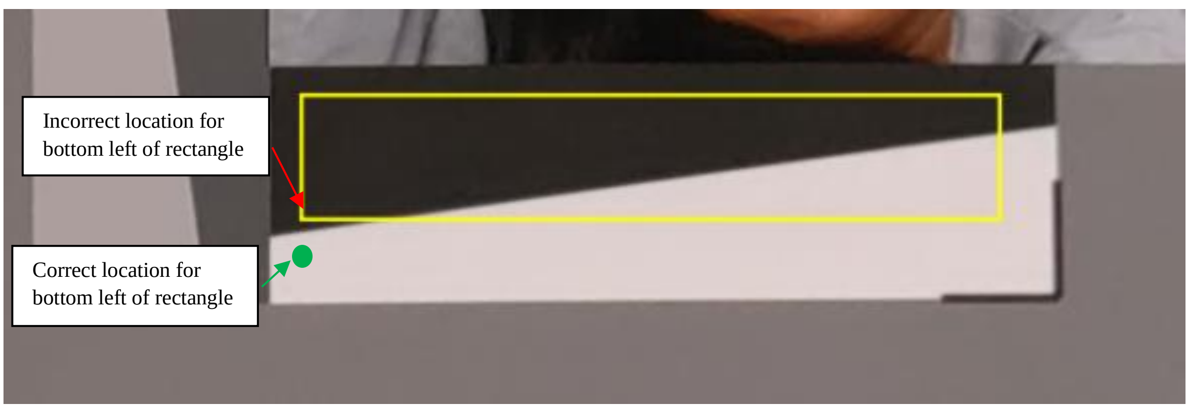


Figure 4-9 *Incorrectly* marked slanted edge. Bottom right of the rectangle lies in black area.

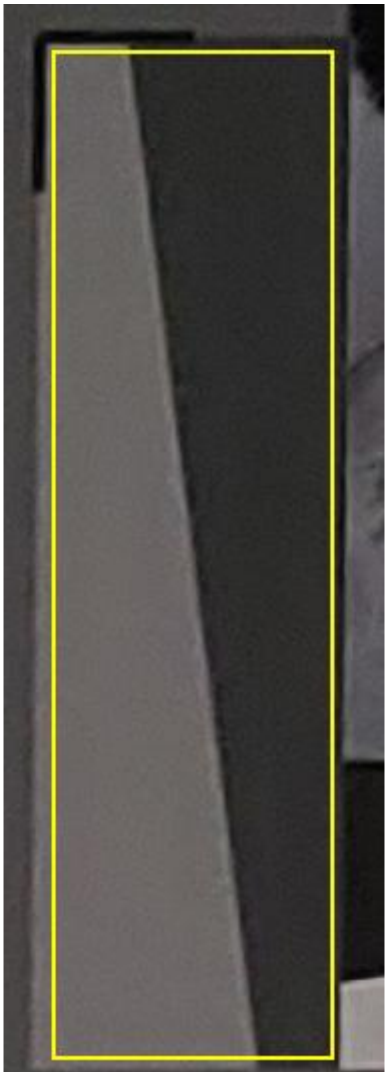


Figure 4-8. Correctly marked low contrast top right slanted edge patch

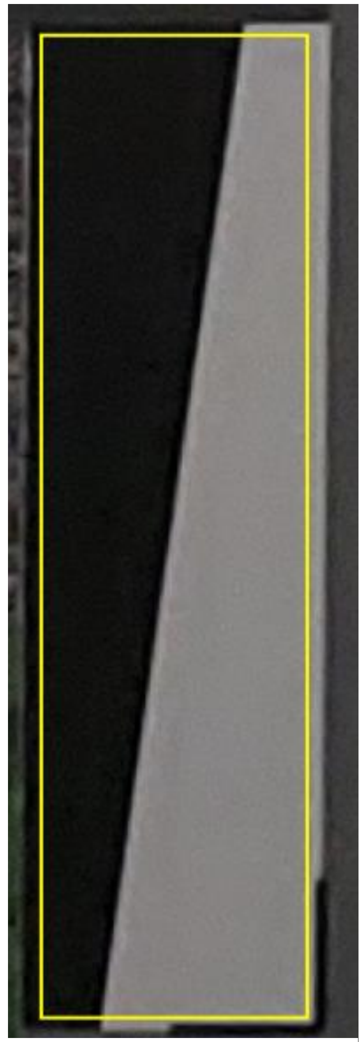


Figure 4-7. Correctly marked high contrast bottom left slanted edge patch

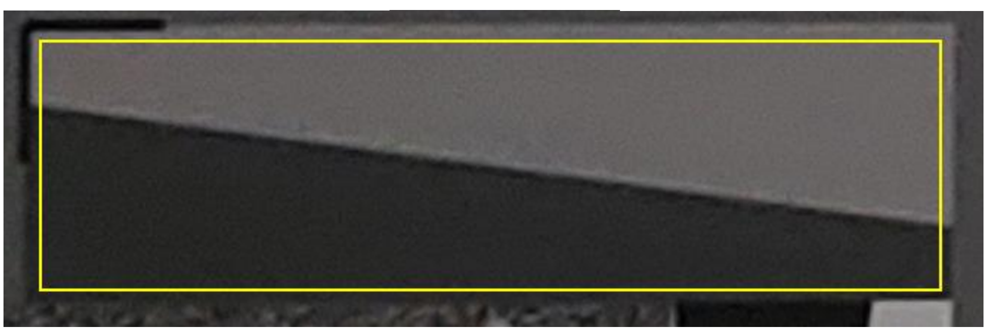


Figure 4-10. Correctly marked low contrast bottom left slanted edge patch

4.2.3 High contrast dead leaves

Mouse right click select “Select dead leaves chart”, mark the high contrast dead leaves by drawing a rectangle as shown in Figure 4-11. Make sure the four markers at the corners of the dead leaf chart are inside the rectangle.

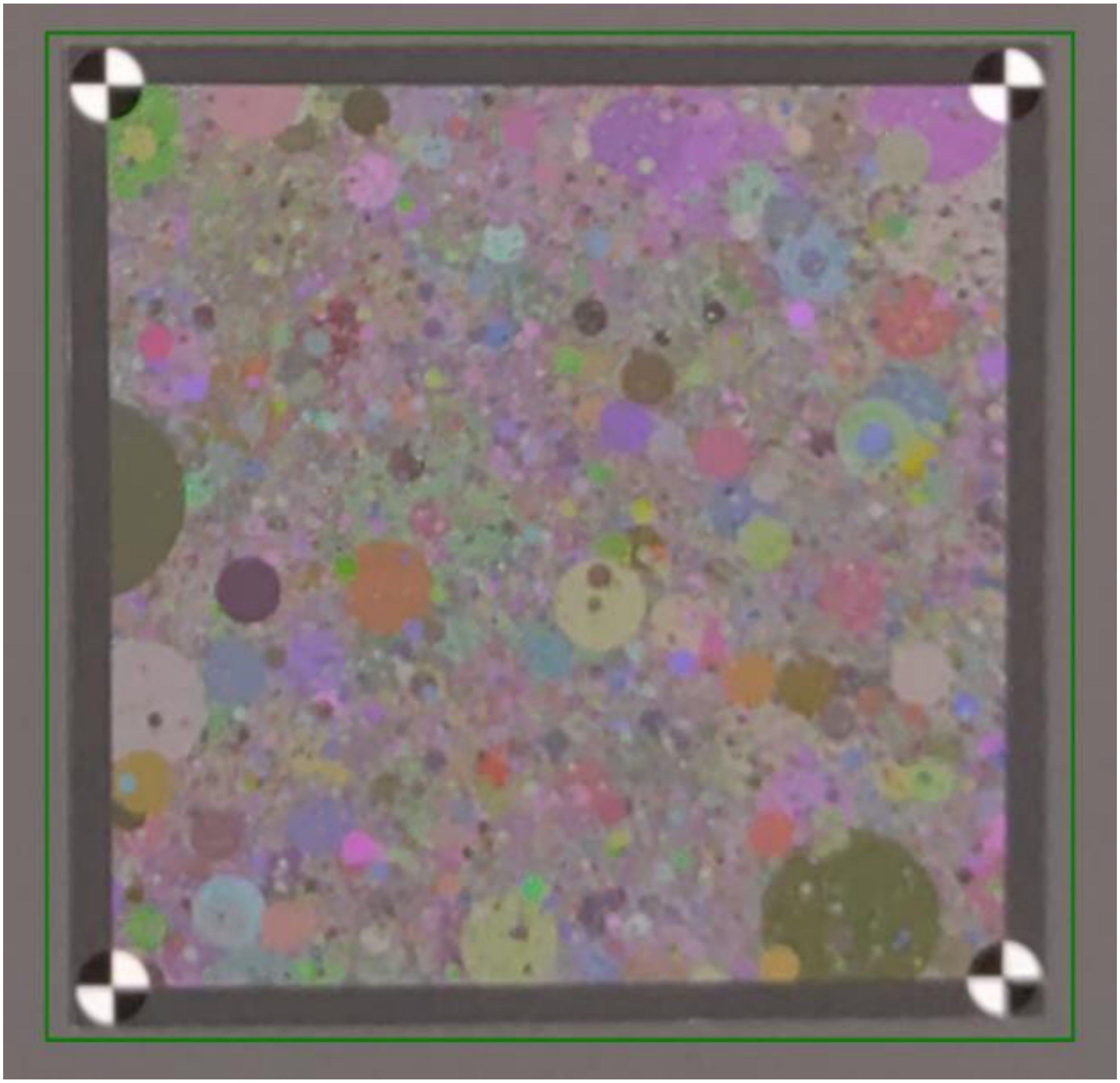


Figure 4-11 High contrast dead leaf

Mouse right click select “Select plain gray area”, draw a rectangle to select gray area to the right of the dead leaves chart. Select as large a region as possible (but ensure that entire rectangle only contains gray patch). This is shown in Figure 4-12

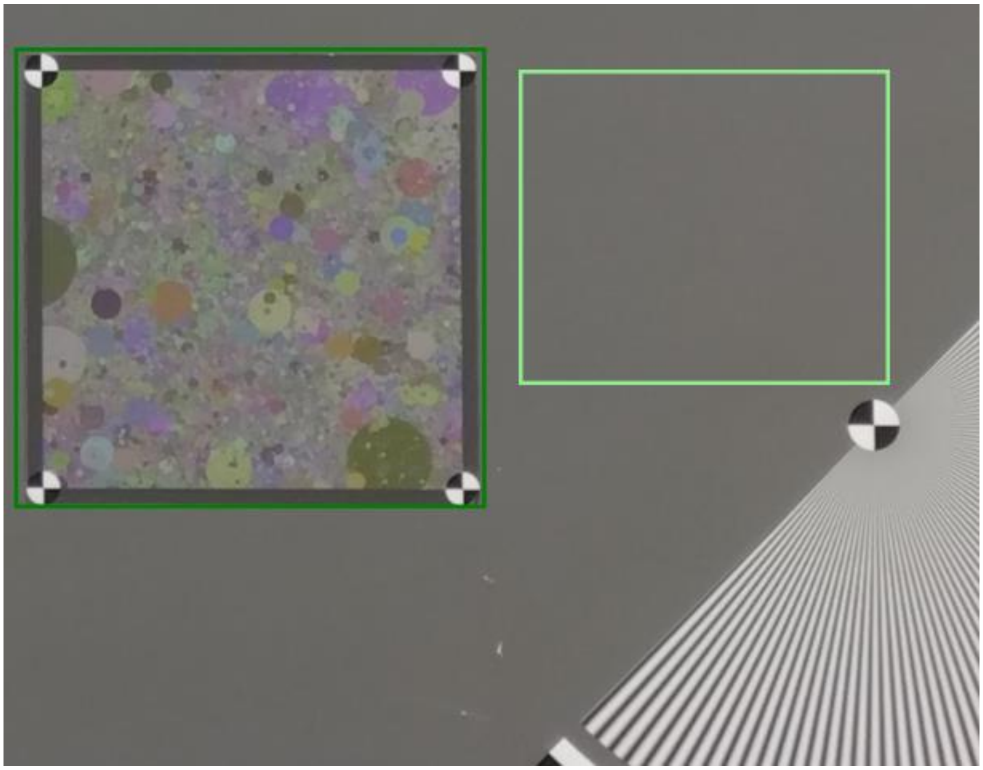


Figure 4-12 High contrast dead leaf and associated gray patch

4.2.4 Low contrast dead leaves

Similar to high contrast dead leaves, mark rectangles for low contrast dead leaf and associated gray patch as shown in Figure 4-13. The gray patch needs to be marked on the left of the low contrast dead leaves.

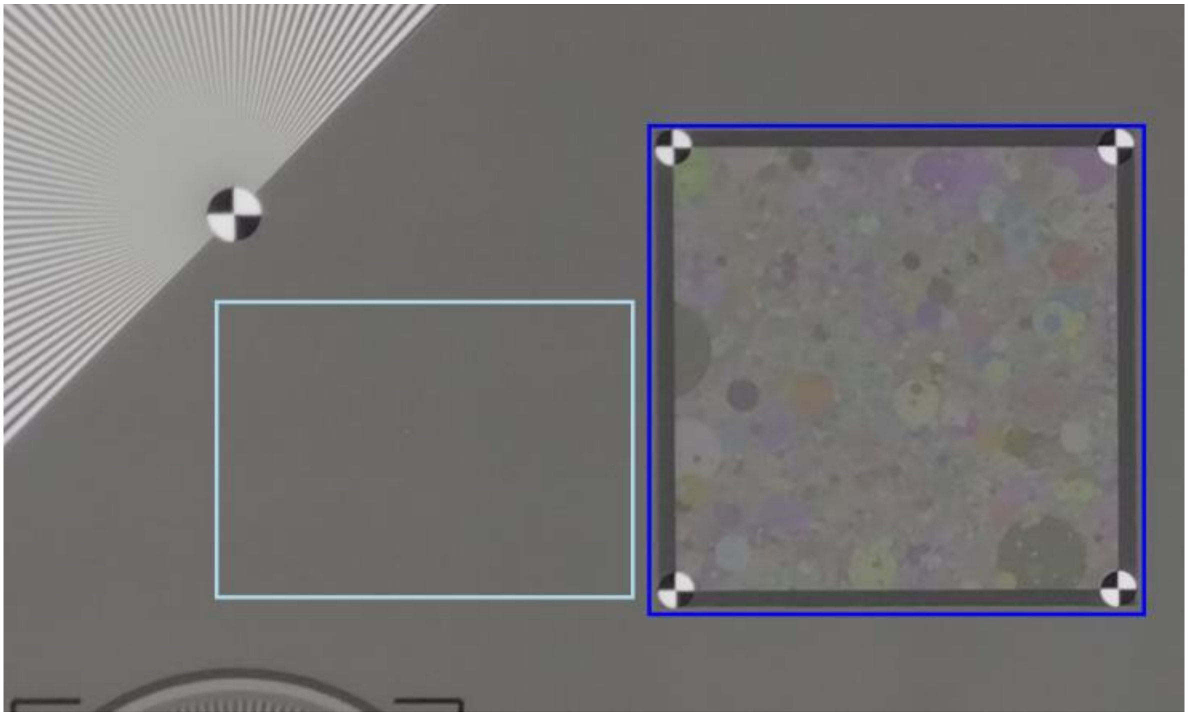


Figure 4-13 Low contrast dead leaf and associated gray patch

4.2.5 Noise patches

Autotune uses 20 luma noise patches located at top, left, right and bottom of the central resolution chart. Marking the noise patch consists of two steps

1. Marking the start locations of the 4 noise patch sets
2. Selecting any one noise patch

4.2.5.1 Select start location

Mouse right click select “Select start locations”, mark the top left of each of the 4 sets of noise patches in this order i) TOP, ii) LEFT, iii) RIGHT, and iv) BOTTOM. This is shown in Figure 4-14

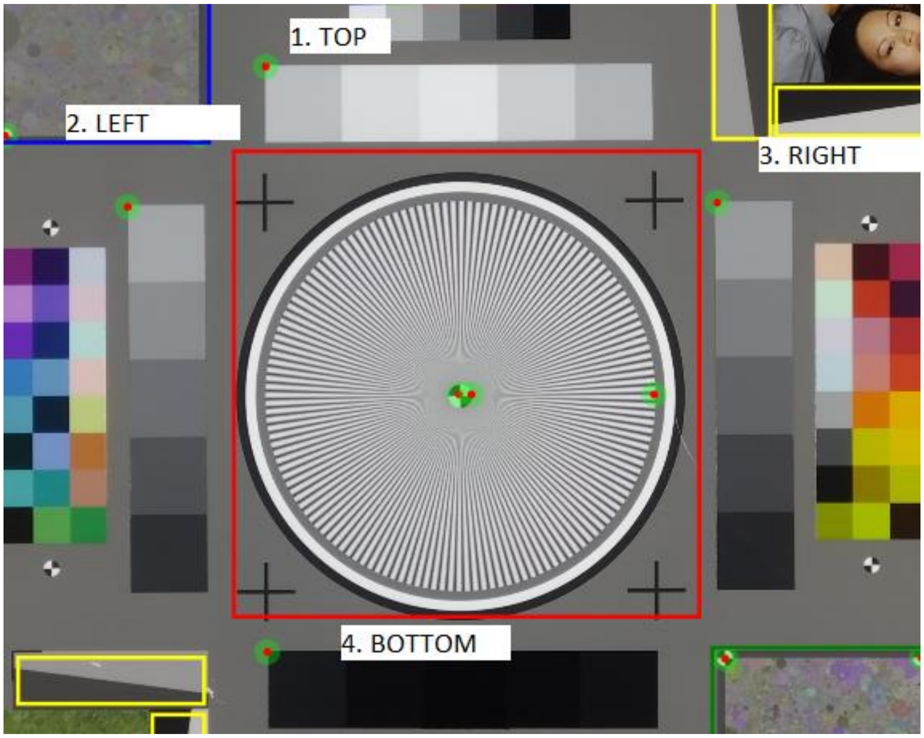


Figure 4-14 Four start location for four sets of noise patches – mark in following order top, left, right, bottom

4.2.5.2 Select one noise patch

Mouse right click select "Select one noise patch" Mark a rectangle to select any one noise patch. Please mark as large a region as possible (but ensure that selected region is completely inside the noise patch). This is shown in Figure 4-15.

Note: recommended that the noise patch is marked in the left or right noise patch strips, as these are easier to mark.

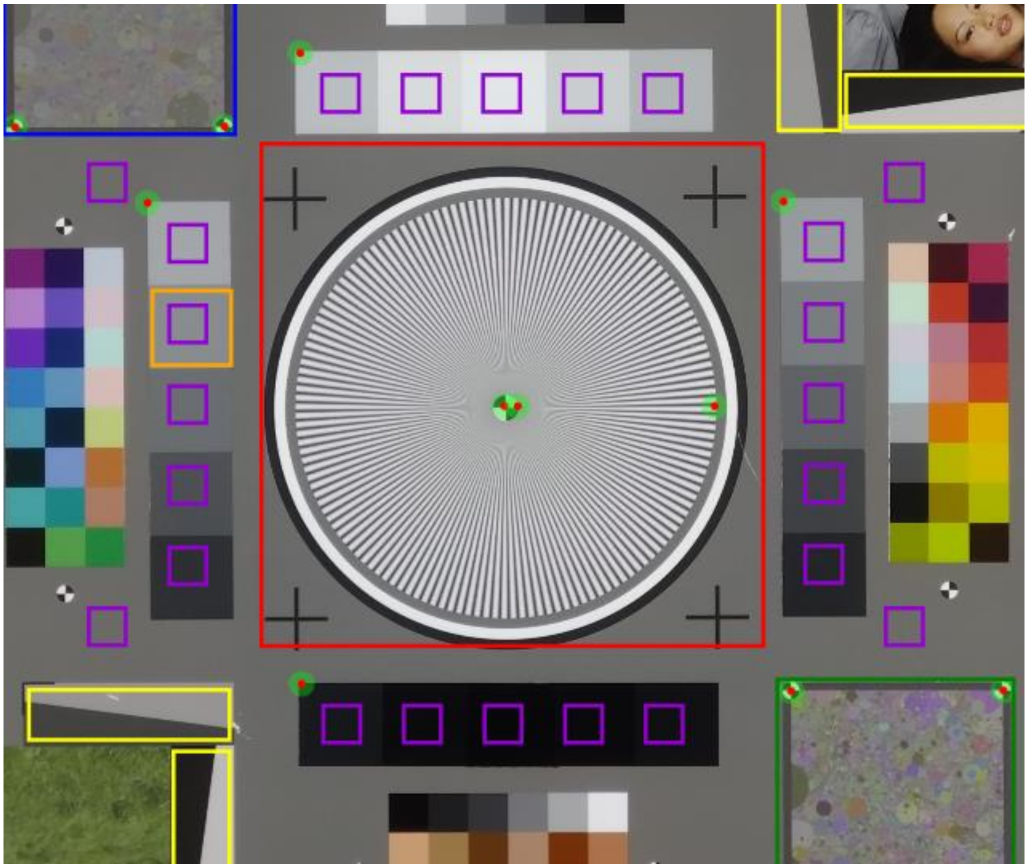


Figure 4-15 Mark one noise patch – select as large an area within any one of the 20 noise patches

4.2.6 Final patch marked image

The final patch marked will look as shown in Figure 4-16.

You will be prompted to save the patch property XML.
This XML will be used for refinement of the low contrast and high contrast dead leaves in the next step.

Figure 4-16 Final patch marked image

4.3 Refinement

4.3.1 Low dead leaves

Click on low dead leaves radio button under “Refine” section at top left of the UI. This will bring up prompt to load a patch property XML file. Please load the patch property XML you created in Section 4.2

This will bring up the below window (Figure 4-17).

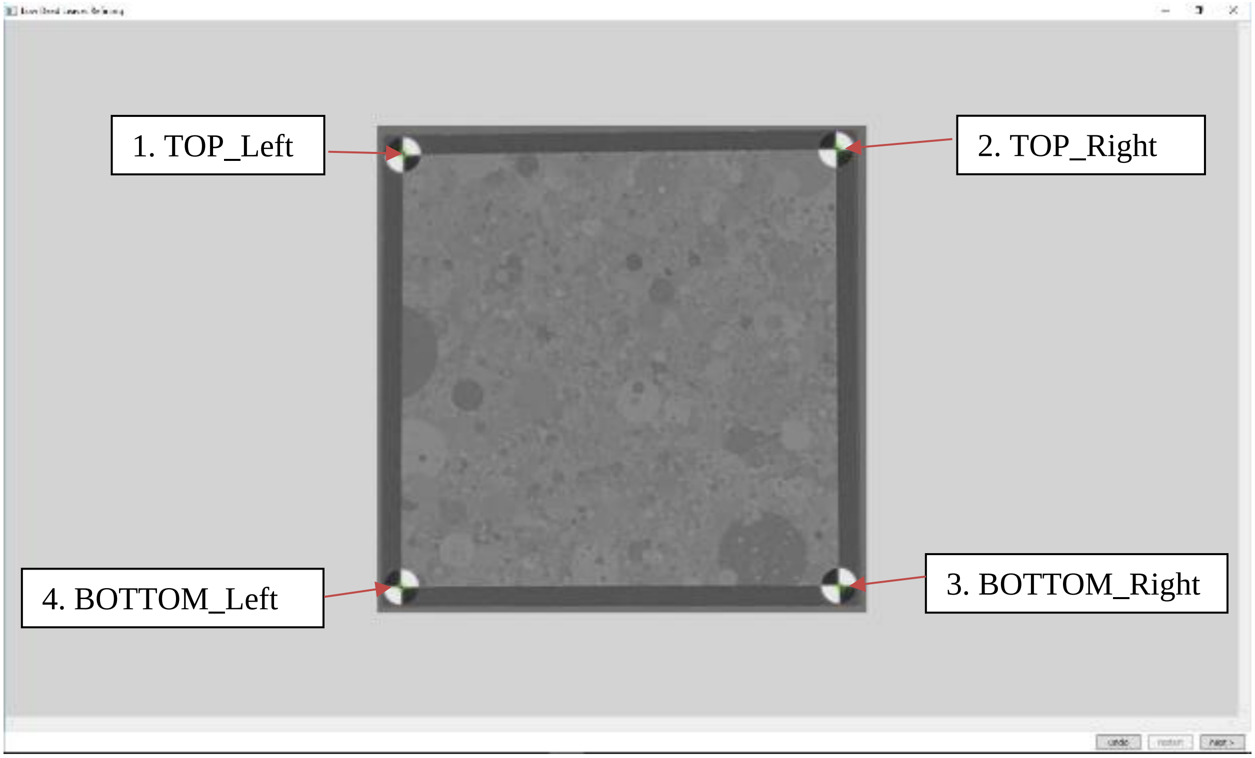


Figure 4-17 Low dead leaf refinement – Step 1

Step 1: Mark the center of the four markers at corners of dead leaf (shown as green dot in the figure) in following order i) Top_Left, ii) Top_Right, iii) Bottom_Right iv) Bottom_Left. (Note that the tool tip will indicate which marker you need to mark).

Click next

Step 2: Mark the center of the four markers again in the new window.

Click next

A new window will show the low contrast dead leaves images warped and projected onto the reference low contrast dead leaves images. If the marking has been done accurately you will *not* observe a green or pink in the image. If not done accurately a green or pink tingle will be present. If tinge is present (see Figure 4-18) click restart and repeat steps 1 and 2, else click end and proceed to High contrast dead leaves.
See Figure 4-19, the tinge is significantly reduced compared to Figure 4-18

This needs to be repeated until the greenish/pinkish tingle is minimized.
Note that it might not be possible to completely eliminate the tinge, repeat the steps until it is minimized.

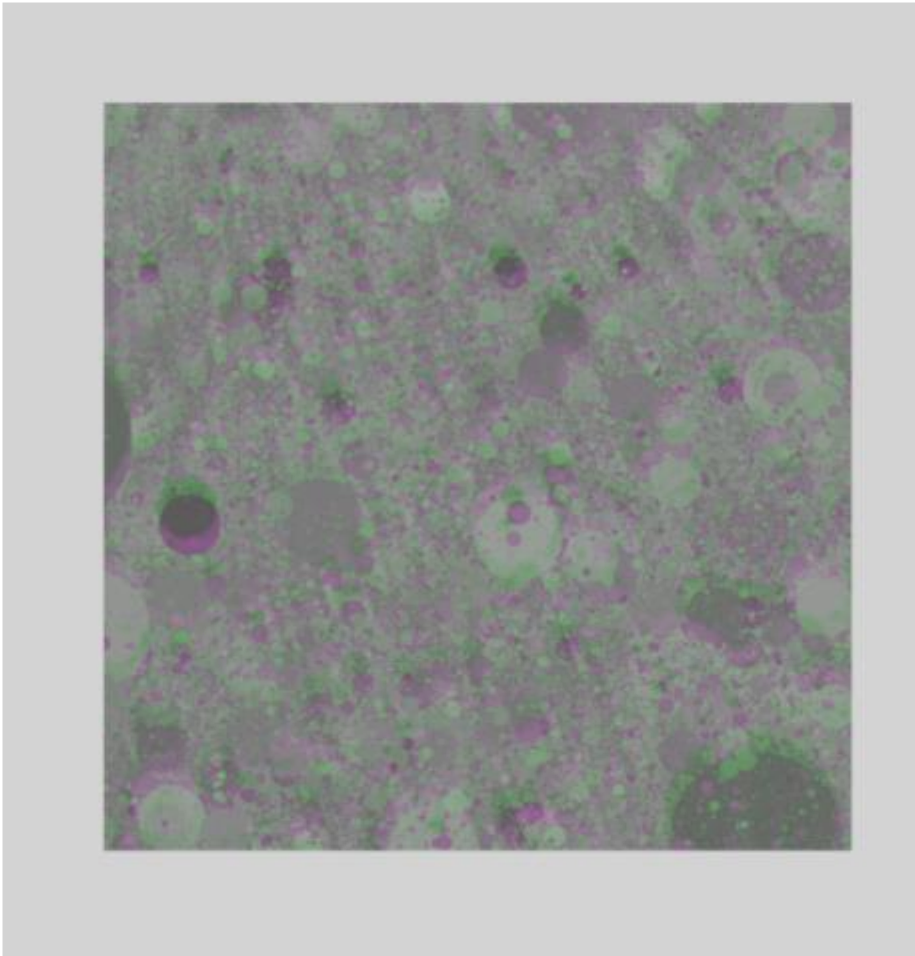


Figure 4-18 Incorrectly warped image greenish and pinkish tinge

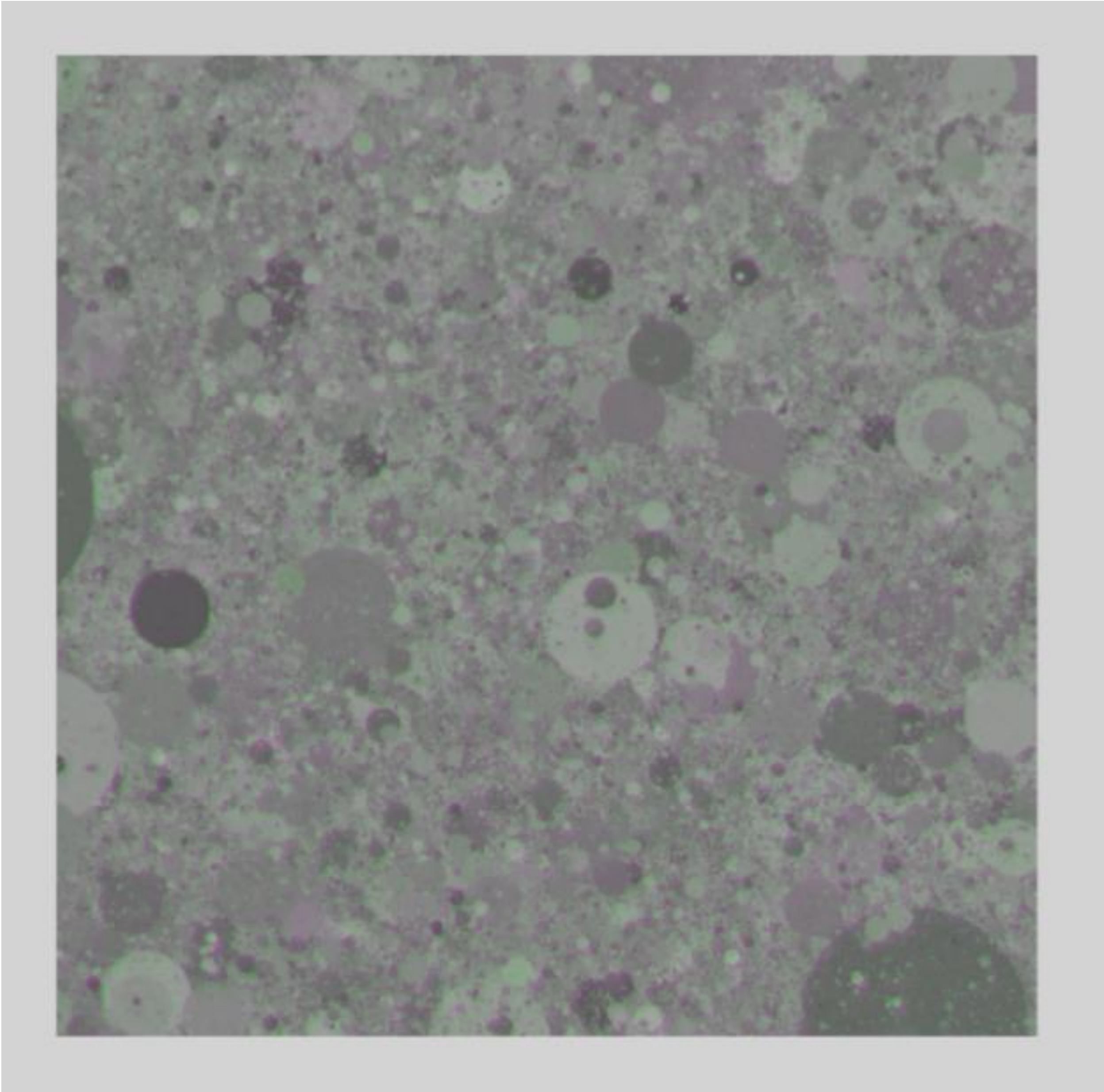


Figure 4-19 Better marking than Figure 4-18. Tinge is much lower.

4.3.2 High contrast dead leaves

Once low contrast dead leaves is completed, the same procedure as in 4.3.1 needs to repeated for high contrast dead leaves.

After completion of high contrast dead leaves click on DONE under refine and save the final patch marked XML.