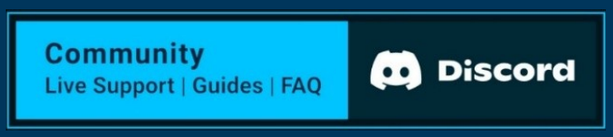
MotionCam PRO
RAW VIDEO CAPTURE

FULL USER GUIDE

4.0.5-Pro Release

Manual Version: 0.0.1

https://discord.gg/Vy4gQNEdNS



https://www.motioncamapp.com/

Contact Us: motioncam.app@gmail.com

USER INTERFACE BASICS (QUICK START 1/2)

DIRECT PREVIEW MODE: Tap toggle DP mode On/Off. You may also hold Vol + button to trigger

HISTOGRAM QUICK TOGGLE: Tap to toggle Histogram On/Off

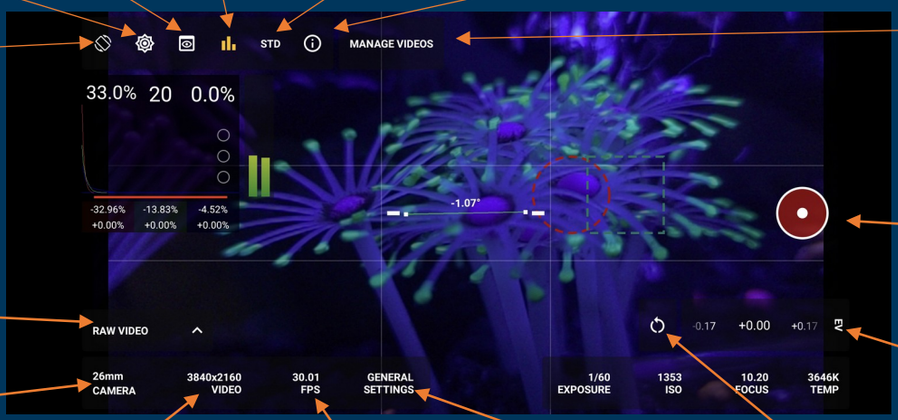
VIEWFINDER RESOLUTION MODE: Press to alternate the preview resolution from STD = Standard (Current selection) to MAX = Maximum, useful for visual granular focusing.

ABOUT: Displays information about MotionCam, including app resources and contact us info

BRIGHTNESS: Tap to toggle (Full vs System setting)

MANAGE VIDEOS: Opens the MCRAW (MotionCam RAWs) file manager. Permits batch handling and provides access to in-app CinemaDNG exporter (Export DNG) as well as single image extractor (Edit Photo) and Video Renderer (Edit Video) modes

ORIENTATION: Tap to rotate, hold to unlock



RECORD BUTTON: Tap to Start/Stop recording. Hold to record with buffer mode

MODE SELECTION: Displays current mode. Tap to open list of modes

EXPOSURE COMPENSATION VALUE: Slide left or right to decrease or increase the exposure compensation amount.

CAMERA & VIEWFINDER SETTINGS: Opens list of device lens IDs and allows for viewfinder adjustments. Swipe left or right to quickly change lenses

CAPTURE SETTINGS: Allows adjustment of capture settings such as resolution crop/binning, RAW stream configuration, storage designation, and more

FRAMERATE: Opens FPS selection (Note: Unsupported values may still work)

GENERAL SETTINGS: Opens General Settings menu options (Histogram, Grid lines, preferences, etc.)

RESET CONTROLS: Press to reset all exposure, metering, focus and white balance settings back to full auto mode
GENERAL SETTINGS: Opens General Settings menu options (Histogram, Grid lines, preferences, etc.)

USER INTERFACE BASICS (QUICK START 2/2)

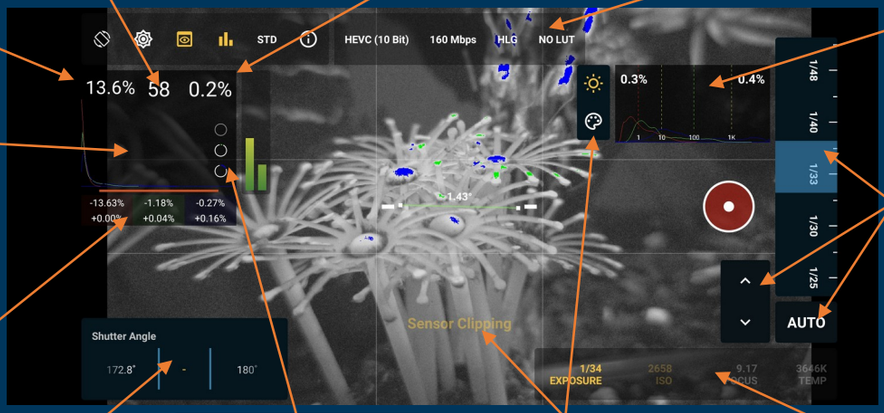
LUMINESCENCE: Quantifies overall amount of light hitting sensor. 0 indicates complete darkness and no light, 1000 represents total white on full sensor area. Great for scene and exposure analysis

CLIPPING PERCENTAGE: Indicates percentage of total pixels reading fully white (Highlight data loss)

DIRECT LOG SETTINGS: Shows current Direct Log mode capture settings. Tap any to open menu or hold LUT to toggle LUT chosen on/off

CRUSHING PERCENTAGE: Indicates percentage of total pixels reading fully black (shadow data loss)

ENCODER HISTOGRAM: Provides histogram data specific for the selected Direct Log settings as they will appear after encoding. Shows crushing and clipping percentages



RAW HISTOGRAM: Tap to expand. Shows sensor RAW data histogram (Not JPEG). Provides precise measurements of light captured. Left = Crushed Shadows. Right = Clipped Light. Vertical bars indicate audio input (Higher bar indicates louder).

CONTROL SLIDER AND TOGGLES: Provides slider for granular controls and arrows for jumping between settings. Press AUTO to undo manual setting. Adapts based on Exposure, ISO, Focus or White Balance/Temperature

INDIVIDUAL CHANNEL CLPPING/CRUSHING: Indicates individual color channels (Red, Green and Blue) and their percentages clipped (positive) and crushed (negative). Ideal for highlight recovery gauging

SPECIAL CONTROL MENU: Provides options for Shutter angles, ISO quick presets and focus racking. Changes based on active selection (Exposure, ISO, focus)

CLIPPING TRAFFIC LIGHTS: Colored indicators for individual channels clipping (Red, Green and Blue). Off indicates no clipping, circle filling up indicates +0.01% of channel clipped and fully circled and lit up indicates +1.00% or more clipped

DIRECT PREVIEW OVERLAYS: States the current DP mode engaged; sRGB/Rec.2020, False Color (Paint palette) or Sensor Clipping (Sun)

SHOOTING PARAMETERS: Shows the Exposure/Shutter Speed, ISP, Focus and White Balance in use. White = Auto and Yellow = Manual. Long press category to lock

Introduction

Welcome to the world of MotionCam Pro! You've now taken the red pill of mobile cameras and are about to embark on a journey that will challenge everything you thought you knew about filming on a phone.

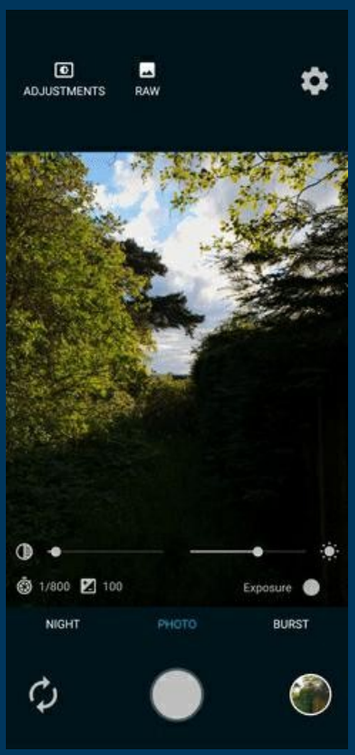
This guide will cover every aspect of the app, from basic operation, user interface, getting started, how to handle outputs, to the technical details behind its capabilities. We'll strive to fully answer most common questions, address all potential issues you may encounter, and explain exactly how things work. Our goal is to give you comprehensive understanding of the app, all available in a one-stop location so you can confidently dive into the world of RAW and advanced mobile videography!

While we will cover most of the essentials, this guide won't delve into complex topics like color grading or ‘best settings’ in DaVinci Resolve, nor the specifics of third-party mods and enhancements relying on root access (which are often popular with the app). These topics are highly complex and depend on individual workflows, preferences and even device models. However, MotionCam Pro stands on its own and is designed to harness the full potential of your device's camera, with or without these external enhancements or programs.

Anyhow, you’ve arrived at the Mt. Everest of Android camera apps! I’m Armando, better known as Ragusaucy, your MotionCam Sherpa! We’ll start by covering what the app is, what makes it special, then cover important basics and technicals. We’ve got a crash course for newbies too if you are new to this! Afterwards, we'll hit the ground running with usage and UI! Let’s get started then!

Our Origins

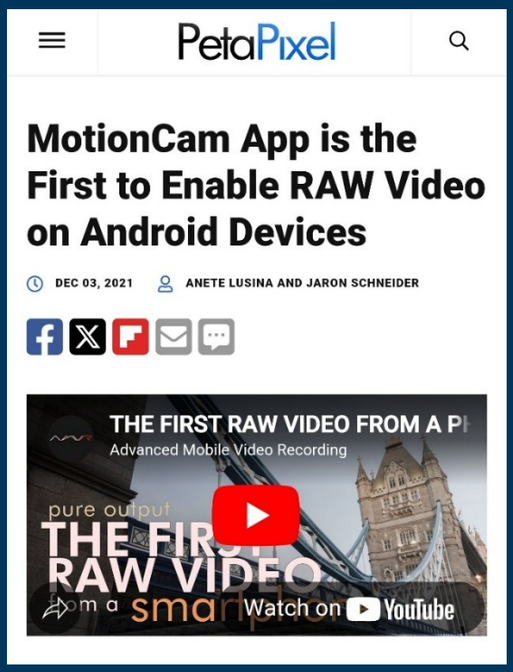
A highly skilled developer, Mirsad, was inspired by Google Pixel phones using multiple frames to create HDR photos. He created a camera app that used a similar technique of harnessing a stream of RAW images, and aimed to improve upon it by avoiding shutter lag and blur during movement. The app continuously captured a burst of RAW images from the same stream used by Google's HDR pipeline and leveraged a motion aware stacking algorithm, resulting in photos with no shutter lag that are resilient to movement, hence it was appropriately called MotionCam.



A visionary power user, Sebastian, having sought for the best possible smartphone camera quality for some time prior, saw the app's potential to capture a burst of RAW images and asked Mirsad to extend the burst, explaining the idea of an app capturing RAW video on a mobile device - an impossibility at the time

As a result of this collaboration, the app achieved a way to capture this burst of RAW images for longer and longer, culminating with the creation of a proprietary, purpose-built format called MCRAW! The app's ability to capture true RAW video was born, dawning a new chapter in Android video! Rest is history!

As the app matured, further functionality was added such as processing MCRAW captures in-app into delivery ready or intermediate log video for non-RAW workflows. The next milestone was reached with the "Direct log" mode which does not record MCRAW, but instead performs RAW image-to-video conversion on-the-fly during capture and encodes directly into options like HEVC or ProRes (and log too) without the overhead of RAW video storage. This is an exceptional ability for an android camera app, since it performs the RAW image processing entirely on its own without relying on the android camera stack or the phone's integrated ISP.



The development of MotionCam Pro is also a story of community collaboration. Development over the app has been very active on platforms like Discord and Telegram, engaging with users to get feedback, troubleshoot issues unique to the android landscape of device variety, and understand the needs of mobile filmmakers, all whom have contributed to the powerhouse that the app has morphed into in the current day. This close relationship with the user base has been crucial to the app's continuous development and its ability to stay at the forefront of mobile video technology.

In summary, MotionCam Pro was not born out of a corporate lab but from a vision of two developers and their perseverance. It's an origin story rooted in a technical challenge (unlocking the RAW power of Android's cameras) and driven by a desire to provide professional-grade tools to a community of creators who were previously underserved, and more importantly about the ability to dream!

About MotionCam Pro…

As you are here, you are already half way to achieving total control of your device's

cameras. You've probably heard all the rave about iPhones and their ProRes Log recording by now, often incorrectly referred to as 'RAW' video. Well, we will show you how to push your Android device to achieve capabilities beyond that! MotionCam empowers users to unlock their device's full camera potential, regardless of their videography experience. Our core mission is to give you absolute control over your phone's imaging quality as well as providing you the tools to achieve it.

Whether you want to capture true RAW video, ProRes Log, everyday video or simply use the app for photography, MotionCam provides comprehensive controls and quality beyond imagination. We believe you'll quickly discover the benefits of this.

What truly sets MotionCam apart is its unique utilization of the Android RAW stream. Typically, Android devices offer developers firmware tools to access camera functions through various image "streams." For example, the photo stream captures JPEGs, and the video stream (also called YUV stream) records compressed video in codecs like HEVC/H.265. You can also capture single RAW images (DNGs) from the RAW stream, which provides the complete and pure sensor data before any processing or compression even takes place; ideal for RAW photography!

MotionCam's innovation lies in its ability to keep that RAW stream open, allowing it to harness and save every bit of uncompressed data generated by the sensor, rather than just a single RAW photo. This is where the magic happens, giving you unprecedented control and visual fidelity over video and photo captures. Forced sharpening? Bad denoising? Dynamic local tonemapping or over saturation? All a thing of the past!

By capturing only from the RAW data source unlike other options on Android or iPhone, our app is capable of fully bypassing your phone’s badly tuned and often undesired image processing that has become associated with phone video capture.

What content to expect ahead…

Ready to begin your journey? We've got a lot to cover, from the basics and fundamentals (if you are new to this, don’t worry, we’ve got you covered!), to the nitty-gritty details, all designed to help you ascend into a true MotionCam maniac like us!

Think of this guide less like a boring washing machine manual and more like a fun "how-to" book for your app experience. Our goal is to walk you through not just the app's many features, but to turn you into a filmmaking force to be reckoned with, all while navigating and learning the intricacies of the Android world. Perhaps by the end of this you’ll even curse towards your phone manufacturer a little bit!

Even if you’re already a pro with a DSLR or Mirrorless camera, stick around! The Android world and smaller sensor capturing have both their own quirks and tricks, and we'll cover all of them (or at least we’ll try our absolute best!) But hey, if you're already familiar with the basics, feel free to skip to the good stuff like the UI, App functions, and the FAQ or troubleshooting sections. Otherwise, we’ll start with basics and essentials!

And remember, even pros needs a little help sometimes! Don’t be a stranger. The MotionCam Pro community is full of amazing people who are ready to lend a hand, some of which I’m happy to call friends, mentors and fellow maniacs! You can find us primarily on Discord, as well as Reddit and even on some discreet Telegram groups. Our community is a huge part of what makes this app great, and many of our users are happy to share their knowledge and expertise. The wealth of expertise, knowledge and skill sets present in our ranks is unparalleled when it comes to any Android Camera app, or many apps altogether. Embrace our RAW brotherhood (or sisterhood, you’re welcome too!)

With that said, let us begin your learning journey!

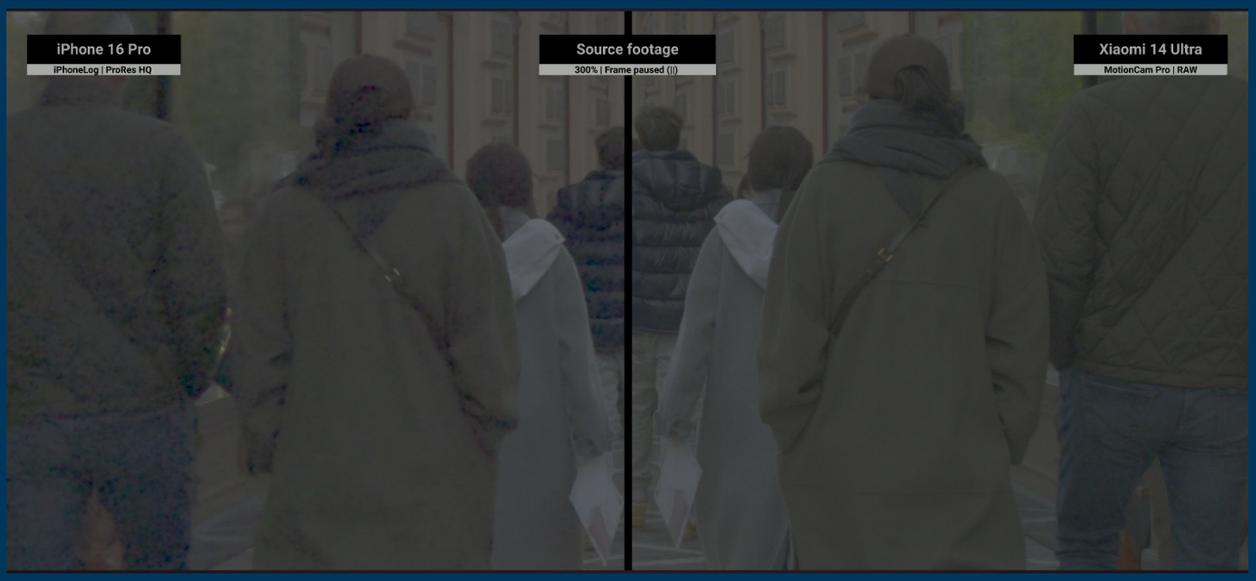


Table of contents [WORK IN PROGRESS]

Fundamentals Crash Course: The Basics

What is RAW video/photography

If you read about all those aspects and remain undeterred, then you have my congratulations! You are part of the 1% in the Android user base. Still feeling confused? Well, let's try to speed run this!

So what's RAW video really? If you are already familiar with all of this, feel free to skip ahead as this course section is for greenhorns, but otherwise, let's demystify this whole "RAW" thing.

Quick heads-up, I'm going to assume you have only a basic understanding of photography at first because if you're not fully familiar with what a RAW image is, you might fall behind quickly. It's totally okay, though! A surprising number of people find their way to MotionCam while searching for the ultimate quality and control, therefore I'm going to do my absolute best to help everyone follow along every step of the way and ensure nobody gets left behind! Let's talk about why we're here: quality.

You've probably heard the term RAW or even DNG (Digital Negative – a RAW file format) thrown around by photographers and videographers (as well as ourselves earlier) who sound like they're speaking a different language. Don't worry, it's not as complicated as it sounds! Think of it this way; a cooked meal vs a bag of ingredients

RAW is quite a fitting name surprisingly. Imagine you're a chef. You've got all these fresh ingredients: a nice tomato, some crisp basil, and garlic. You could just grab a jar of Ragu (haha, see what I did there?) and call it a day. That jar of Ragu is basically a JPEG file. The camera has already chopped, cooked, and bottled everything for you. It's ready to go, but you can't really change the recipe. It's a done deal. It's like saying you want to take out the onions in that jar of Ragu… Good luck! In the same spirit, try changing the white balance of a JPEG, you'll quickly realize it's a bad idea

A RAW file, on the other hand, Is that shopping bag full of untouched ingredients. It's all the original data from the camera's sensor, but it's not a finished image you can share. You have to "cook" it yourself in an editor program like Lightroom for example. You get to be the chef, deciding how much brightness, contrast, and color to add, to name a few of the limitless possibilities. Remember that white balance I mentioned as an example? You can change it after capturing with zero penalties! This gives you ultimate creative control.

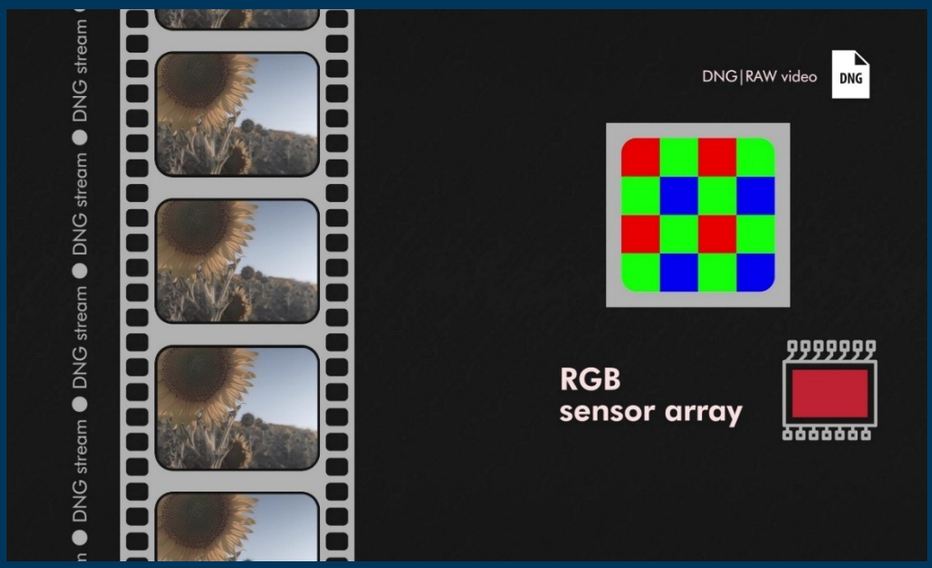
Now of course, if you cook everything fresh, you'll need a whole pantry and refrigerator to store all the uncooked ingredients which take far more space when not ready. In the same spirit, RAW files are much larger than a JPEG.

You also don't open up your fridge to impress your partner with the future lasagna you are gonna cook them, nor do you serve them a frozen meat slab and say look how high quality the wagyu is (if you do, you'll be single soon!)

Essentially, be aware these are not files you can simply open and expect them to look great at first. You will often need special programs to actually see these files too, MotionCam RAW outputs are no different.

It's completely normal for the image to be noisy, soft, flat looking/colorless, or downright underwhelming at first (not always). This will quickly change once you get to cooking however (denoise, grade, stabilize, etc), and a great chef can absolutely make a steak come alive! The same goes for RAW imagery!

Now, what is RAW video? It's quite literally the same! Except instead of a single image, you are now shooting at 60 RAW frames every second with audio for example. All the same benefits of RAW photography apply!



Here's where things diverge from the photo comparison however. With photos, as it's a single file, you can afford to have a decent file size at the end, so you 'compress' a processed RAW image with all but the essential details and make it a JPEG, but with video this is still too much information, so you gotta make it even smaller! Enter the solution…

Encoding/Decoding

Video encoding is the process of compressing a RAW video file into a smaller, more manageable file size, beyond what even JPEG accomplishes for photos. It's essential because RAW video, which is the uncompressed footage straight from a camera, is enormous. Think of it like taking a giant, bag of tomatoes and fresh herbs, then making it into a nice bottle of ketchup. The goal is to reduce the file size without a noticeable loss in quality. It's like saying lol, you encoded 'laugh out loud' into 'lol', and the reader decodes it back into laugh out loud again once they read it. Simple enough, right?

Imagine you have a long sentence you want to tell a friend. You could write it down in three different ways:

No Compression (The RAW, Full, Unedited Sentence)
"I will be there in approximately ten minutes to pick you up so that we can go to the concert together." This is the original message, with every word spelled out. It's the most accurate representation of what you want to say. In video terms, this is like RAW video —huge in size, but with every single pixel of information preserved perfectly.

Lossless Compression (Using Abbreviations)
"I will b there in ~10 mins to pick u up so we go 2 the concert together." Here you simply used common abbreviations. You've made the sentence shorter, but your friend can still perfectly reconstruct the original meaning. This is like lossless video compression, file size is smaller, but all the original data can be recovered perfectly without any loss of quality.

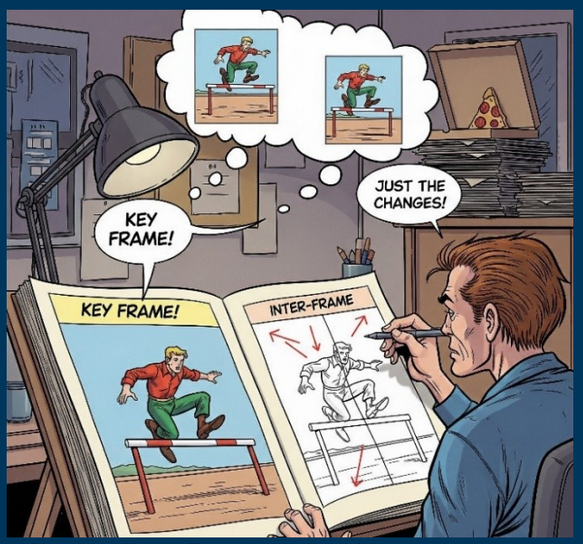
Lossy Compression (Using Slang)
"brb, omw to pick u up for the concert. ttyl." Now you've drastically shortened the message, but you've also lost some of the original information. "I will be there in approximately ten minutes" is compressed into "omw" (on my way). The "approximately ten minutes" part is lost entirely. "so that we can go to the concert together" is also heavily compressed. You have to infer the full meaning, and some details are gone forever.

Will your friend notice enough to care? Probably not. This is like lossy video compression. You get a much smaller file size which is what you may be used to, which is great for streaming or storage, but you permanently lose some of the original data.

You can't get back the original quality once it's been compressed too much. With videos or photos, the human eye might not necessarily notice the missing details, but they are gone forever.

Encoders tend to operate under different rules that they follow for compressing, depending on which use case they are intended for, as well as how advanced or new they are (like generational slangs!)

It's kind of like asking 'w.t.f.' to a polite person, they may think Why The Face? You need to have a standard 'set of rules' to encode (shrink) and decode (unshrink) in a way that can be followed along. These 'rules' or compression languages are called codecs (COmpressor/DECcompressor)

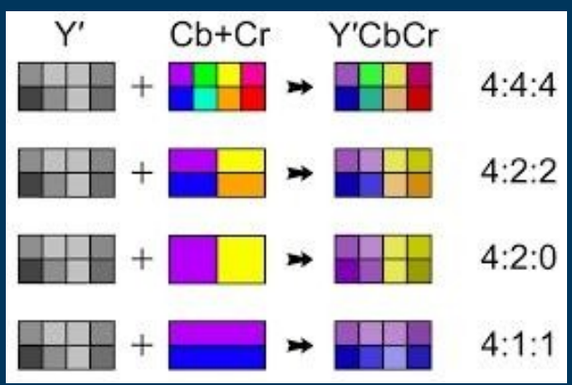


Some codecs are better for editing (Like ProRes) which are not as tightly packed and compressed but give a good enough size reduction without compromising quality. Some are better for final delivery or storage, like H.264/H.265 or AV1, but may be harder to edit because of how compressed they are, as well as they may have compressed away too many important details sometimes.

Video codecs work by getting rid of information our eyes won't miss. For example, in a video of someone talking, the background probably doesn't change much. The codec notices this and saves that background information just once, instead of thousands of times for every single frame. This clever trick makes the file size much smaller. Because a lossy codec throws away some data, it's a trade-off: you get a small, efficient file in exchange for a tiny, often unnoticeable, drop in quality.

Now, if you survived all of this, I'm sorry for the torture, but there's also another aspect, haha! Chroma subsampling… I was unsure if I should have covered this one but the gist of it is a video "shortcut" that reduces the amount of data needed to store or stream a video by sharing color information between nearby pixels to avoid saving it for all of them. It works because our eyes are much more sensitive to changes in brightness than color, so we don't notice the missing color detail.

I won't go beyond here but you'll see some codecs can do 4:2:0, 4:2:2 and 4:4:4, essentially its how densely you compress the colors. If you see someone flex they can record in 4:2:2 and you can only do 4:2:0, joke's on them— RAW video is even better than 4:4:4! It's a detail that's a magnet for forum arguments and that makes the inner spec junkie happy. On a serious note however, using better chroma subsampling can help you with stuff like green screen editing and whatnot but don't stress too much about it if you didn't know what it meant already as that means there's a good chance you didn't need it (unless you are a pro, then you probably know when it's needed)



As far as decoding goes, it's simply about uncompressing all of the above via the same set of rules or language used. Both encoding and decoding may be done via hardware acceleration (an integrated device chip that specializes in encoding/decoding) allowing for efficient recording and playback as well as software acceleration (Completed via brute forcing by CPU) which is much harder but allows for recording or playback even if you don't have a special chip – however it's substantially more demanding and will impact performance more.

Bitrate

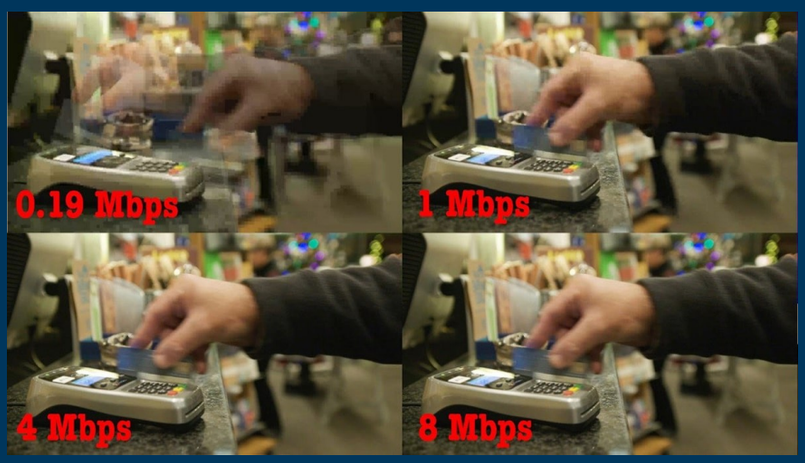
This one's a breeze, I promise! Bitrate is simply the data density of your video—how much information is packed into each second. It's measured in Mbps (Megabits per second, not Megabytes—a very important distinction!). Think of it as the maximum weight allowance for your digital luggage.

Have you ever flown on an airplane? Let's say the airline gives you a codec, your luggage (the set of rules for compressing your video). The luggage's job is to efficiently carry all your "essentials" as a standardized container. The bitrate is your luggage's weight limit. If you have a 30kg limit, you have to stick to it or risk paying a penalty.

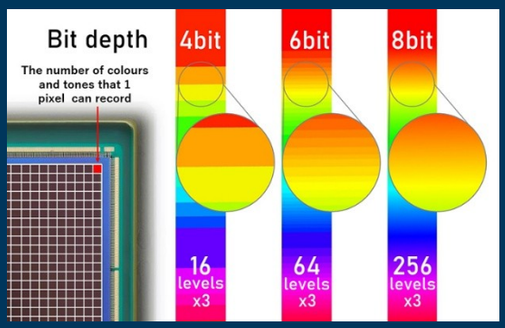
The same goes for your video! Your codec will try its best to maximize data efficiency while staying within the bitrate limit you've set.

Too much bitrate? You get to pack extra things you might have left behind otherwise, like that nice bottle of fancy liquor. The video will look pristine, and while at some point you might hit diminishing returns (you can only pack so much detail!), it's generally a good thing.

Too little bitrate? This is where you have to make tough choices. You might be forced to decide if you can even bring an extra pair of underwear. The quality suffers, and you'll start seeing nasty compression artifacts like blurry smudges, "ghosting" during movement, or blocky blobs in dark areas.
It's all about finding that sweet spot. It doesn't matter if you have higher resolution (a bigger luggage), your weight limit will ultimately restrict you too. It's why some older YouTube videos in 720p looked better than today's 1080p content—they were simply given a higher bitrate allowance, meaning more "data density" and less artifacting. So, choose your bitrate wisely, and avoid a digital baggage claim disaster (Yes, I am a Dad... How did you know??)

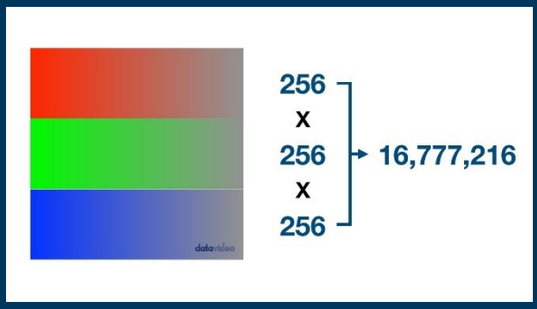


Color depth/Bit depth/White level



8-bit, 10-bit, HDR, SDR, Log… What does it all mean? Well, an important factor about recording is the bit depth of your content or that the sensor output at. Also referred to as color depth, bit depth is a fancy way of saying how many colors your sensor can capture and reproduce.

Think of your camera sensor as a little artist, and its job is to paint a picture of the world it sees. The tools this artist has at its disposal are what we're talking about here.
Bit depth/color depth, is a super-fancy way of saying "how many colors can my camera use?" Imagine you're drawing a beautiful sunset. If you only have a small box of crayons, say 8-bit, your options are limited. The sky might look a little blocky, with a few chunky transitions from orange to red. You've got 256 shades of each primary color (red, green, and blue - so 256 x 256 x 256), which gives you over 16 million possible colors in total. That sounds like a ton, right? It is! But believe it or not, our eyes are so good that they can still see those little "steps" in the color gradient in particular if you push the edit too hard or compress too much, especially in smooth skies or subtle shadows. This is called color banding.



Now, if you upgrade to a bigger crayon box (a 10-bit sensor) you get a whopping 1,024 shades for each color. That's over a billion possible colors! With all those extra crayons, your sunset gradients are going to be buttery smooth. The extra colors fill in those gaps, giving you a much more lifelike and detailed picture. This is the superpower of higher bit depth: it captures and reproduces richer colors and smoother transitions.

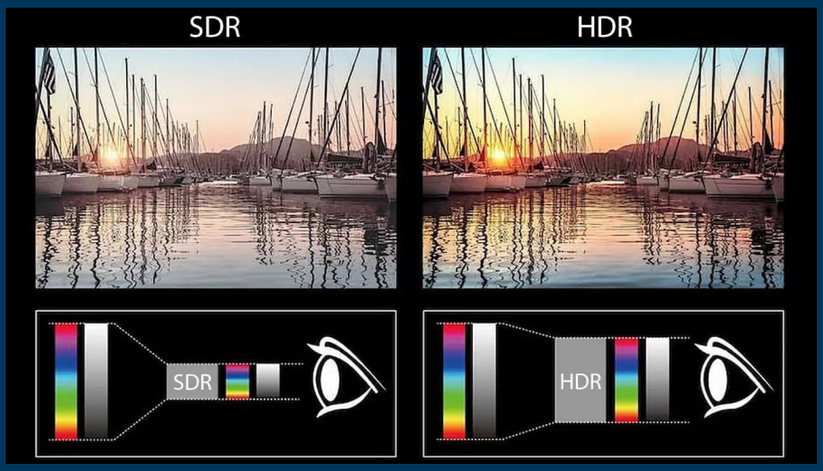
To summarize, 8-bit is 256 shades per channel, 10-bit is 1,024 shades, 12-bit is 4,096 then 14-bit is 16,384 and so on.

Essentially this allows you to capture more data and reproduce it as well. As far as capturing goes, more never hurts, but of course more data means more storage is required!

When it comes to the final product, your video will be in

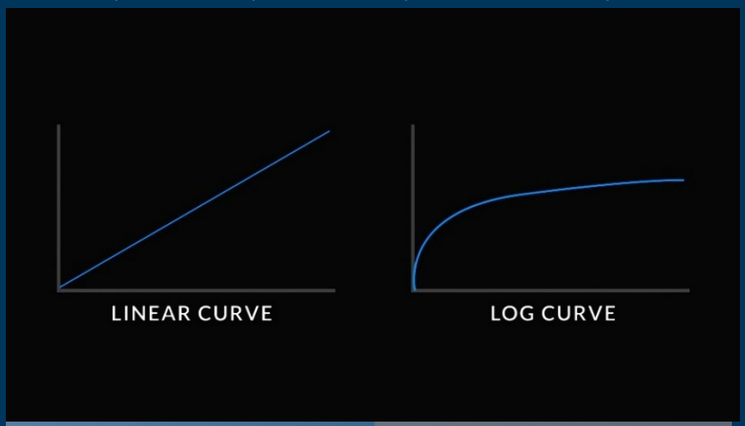
one of two flavors: SDR or HDR. SDR, or Standard Dynamic Range, is the classic, everyday video format we've been using forever, which is usually 8-bit! Think of it as your regular TV show. It's designed to look good on almost any screen and is easy to deliver over the internet. To give you an idea, most of the videos you see on YouTube are in SDR because it requires less data to stream, and everyone can watch it without any issues. It's a reliable workhorse. If you don't believe it, search any video and open the resolution settings on YouTube and search for the 'HDR' tag, you'll quickly realize how uncommon it remains.

High Dynamic Range, is like an upgraded version of SDR. It leverages higher bit depths (like 10-bit) to capture and display a wider range of brightness and colors. This means you can see more detail in the super-bright highlights while rarely surpassing dynamic range in delivery. It's like the difference between looking at a photograph of a sunset and being there in person, squinting at the sun while still being able to see the details in the shadows below. However, delivering HDR content is tricky. Not only are the file sizes larger (remember the luggage example?), but the screen you're watching it on has to be able to display it properly. If a display isn't HDR-capable, your content will appear lacking in contrast and saturation, which is why you don't see HDR everywhere yet—it's still a bit of a fancy, high-maintenance friend best reserved for professional projects and those with the right gear. Don't be allergic to 8-bit/SDR exporting! You'd be shocked at how many people feel using it is a bad idea.



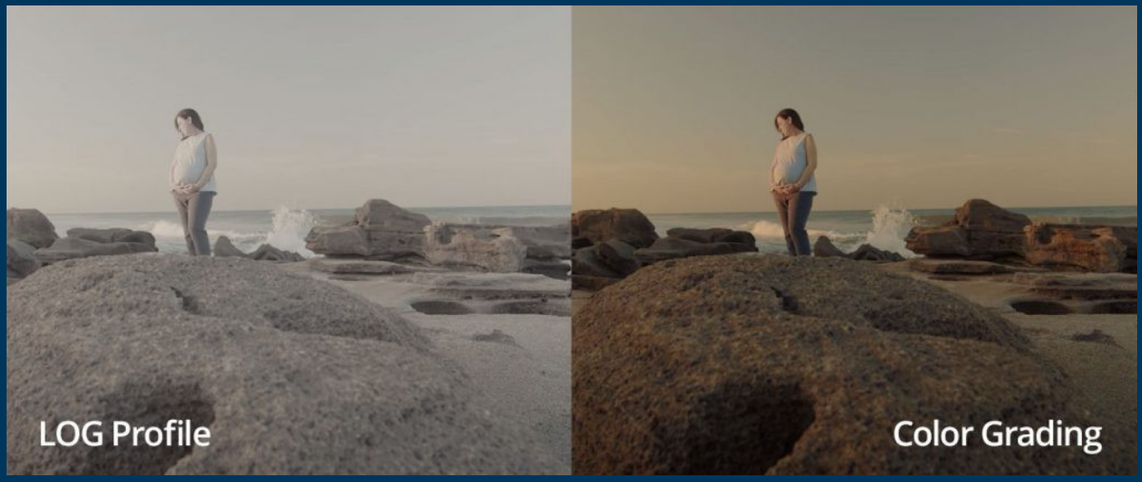
Now I hear you thinking, why shoot in 10-bit if you'll export in 8-bit? Well, it's similar to shooting in 4K to export in FHD 1080p, in this example you could afford more cropping or reframing while maintaining more details on you final edit that would have been lost had your initial capture been in 1080p. Editing latitude! You also get to control the tone mapping process better. Tone mapping is the crucial process that distributes the wide range of brightness from a high dynamic range camera capture onto a standard display. This ensures that you can see detail in both the brightest highlights and the deepest shadows, preventing clipped whites or crushed blacks that exceed your display's limits (exceeding its palette reach)

So, you've decided to shoot in 10-bit to get all those lovely colors. But what about Log? Think of Log as a special way of packing your video data to protect it. When you record normal encoded video, the image colors are generally already in a way to that may look good right out of the camera. But this can sometimes impact the deep colors out the very bright or very dark parts of the image during encoding, losing some of that precious detail in those areas.



Log, on the other hand, is a recording mode that flattens out the colors and contrast on purpose. The resulting footage looks a little gray and washed out, which seems strange, right? But it's actually a brilliant trick! It's like putting a "fragile" sticker on your luggage, but instead of the airline being careful, this sticker tells your editing software, "Hey, this luggage is full of precious data, so don't compress it too hard!" This flat look gives you a ton of flexibility later on. When you get the footage into your editing software, you can "de-log" it using a special filter called a LUT (Look-Up Table) or by simply telling the software what camera setting you used. This brings the colors and contrast back to life, but you now have way more room to play with the image; adjusting brightness, colors, and shadows without the picture falling apart.

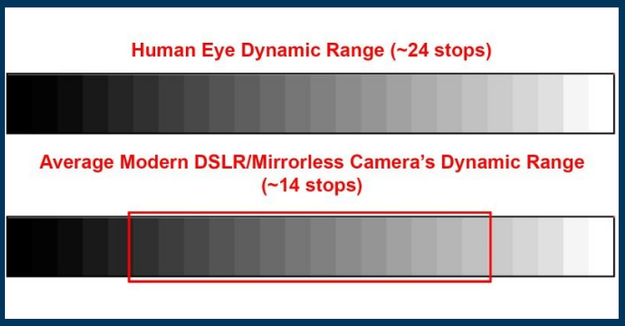
While it's no RAW video by any means, Log does preserve more data than standard delivery methods, while keeping a notable size advantage. While it's possible to shoot Log in 8-bit, it's generally not recommended because you'll lose a lot of that rich data you were trying to protect in the first place! The magic of Log really shines when it's paired with a 10-bit (or above) sensor. And don't worry! Most phone cameras nowadays almost are guaranteed to have a 10-bit sensor! More on that later.



Dynamic Range

You've reached the end of the crash course! Before we jump into the app, let's cover one of the most critical aspects of cameras that you'll have to deal with every time you shoot. FYI, we'll be talking a lot about stops (1 stop = doubling or halving of light intake, whether through ISO, exposure or aperture – eg. Going from ISO 500 to 1000 = +1 stop)

Dynamic range is a camera's ability to capture detail in both the brightest and darkest parts of a photo at the same time. Let's think of a scene's range of light, from the deepest shadows to the brightest highlights, as a piano. A scene with a very wide dynamic range is like a huge, expansive piano with many, many keys, representing a vast range of tones. A scene with a narrower dynamic range is like a smaller piano with fewer keys.



Now, your camera's dynamic range is like the reach of the pianist's arms as they sit at the piano. A camera with a limited dynamic range has a pianist with shorter arms, only able to comfortably reach a certain section of the keys. A camera with a wider dynamic range has a pianist with longer arms, able to play across a broader section.

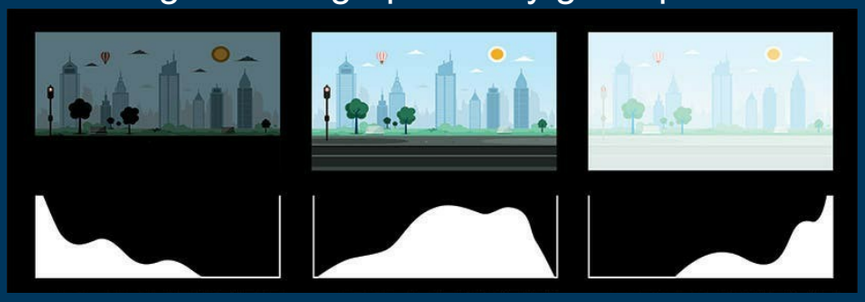
The human eye is like a pianist with incredibly long arms, able to perceive and distinguish almost all the keys on even the biggest piano. However, a typical modern camera's "arm reach" (dynamic range) is more limited, usually capturing around 12-14 stops of light. This means that when faced with a "very big piano" scene (high dynamic range), the camera can only capture the details within the reach of its "arms."

When the "piano" (the scene's dynamic range) is wider than the "pianist's reach" (the camera's dynamic range), you face a choice. This is like the pianist's thoughts, deciding how to prioritize their limited reach. If you want to make sure the dark areas have detail, you'll need to expose for the shadows. This is like the pianist focusing their reach on the lower keys to capture the deep bass notes. However, this risks "clipping" the highlights, where the brightest parts of the scene turn into pure white with no detail. Alternatively, you can expose for the highlights to preserve detail in the bright areas. This is like the pianist shifting their reach to the higher keys to capture the bright treble notes. But this risks "crushing" the shadows, making the darkest parts of the scene become pure black with no detail.

Understanding that your camera has a limited "arm reach" compared to the vast "piano" of the real world, and learning how to choose which part of the "keyboard" to prioritize through exposure, is crucial for capturing the image you intend.

Now you are probably wondering, can't I increase my reach somehow? The answer is – it depends. Bigger sensors generally have more dynamic range, with phones being around 11 stops give or take and a Full Frame about 14-16 stops. Phones can apply HDR merging in which they can sometimes, via hardware methods, capture multiple exposures simultaneously during video, and blend them. We won't touch on this yet (*Cough* DCG… *Cough*) but it is indeed possible to extract such power with MotionCam. Some phones do this, like iPhones, but since they often use the different (inferior) methods, it can introduce visual artifacts like ghosting and whatnot. Nevertheless, it is indeed possible to go head to head with a much bigger sensor.

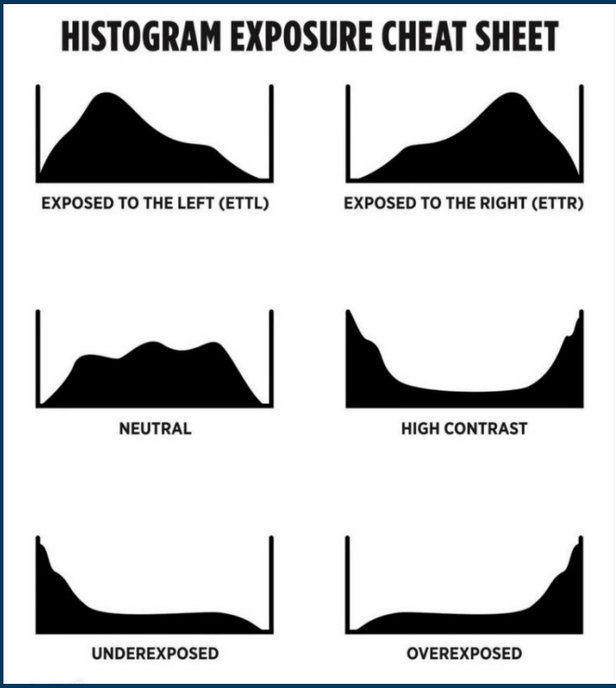
Back on track, how do you know if your "pianist's arms" are long enough for the "piano" you're trying to play? This is where the histogram comes in. A histogram is a graph that visually represents the tonal values in your photo. The left side of the graph represents the shadows (the low notes), the right side represents the highlights (the high notes), and the middle represents the mid-tones. The height of the graph at any given point indicates the number of pixels at that specific brightness level.



Think of a histogram like a soundboard or a visualizer for the piano example. If the graph is all clustered in the center, it means your scene has a very narrow dynamic range, like a small piano with only a few keys. If the graph is spread out across the entire horizontal axis, it means your scene has a wide dynamic range, like a big piano.

When a histogram touches the left edge, it indicates that the shadows are "crushed"—there is no detail in the darkest areas. The pianist's arm couldn't reach the lowest notes, so all those details are lost to pure black. When a histogram touches the right edge, it indicates that the highlights are "clipped"—there is no detail in the brightest areas. The

pianist's arm couldn't reach the highest notes, and all that information is lost to pure white.

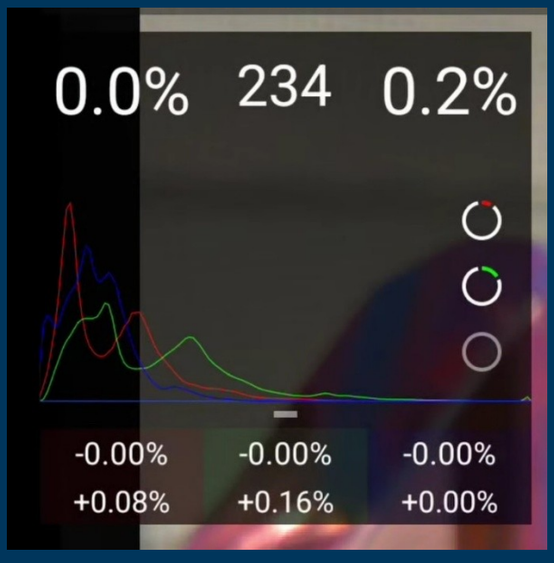


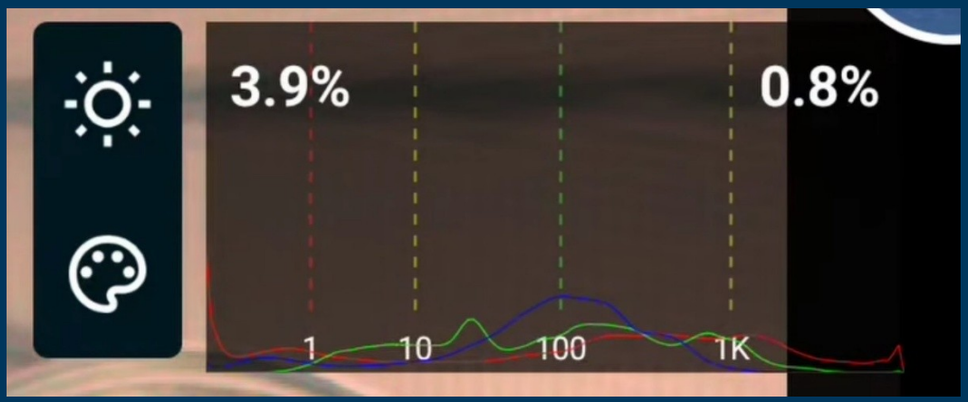
By learning to read the histogram, you can quickly assess the dynamic range of a scene and decide how to best expose your image to capture the most important details, avoiding either crushed shadows or clipped highlights.

We'll touch more on the histogram during the UI section, but rest assured, with MotionCam we've got one of the best ones available in modern cameras! A RAW + Video exposure histogram each, since as stated prior different strategies apply! More on that later, but for now, enough of the theoretical – you've now officially graduated the Ragusaucy Foundational Crash Course!

I do want to give a disclaimer, some aspects have been heavily oversimplified to ensure basic initial grasp of these concepts. As we ramp up, we'll provide the correct and in-depth technical explanation on these topics for you to achieve true mastery of your outputs, but for now, we'll start with this! If you fancy more hard-core content, we've got your back as well! Consider this your departure from Everest Base camp!

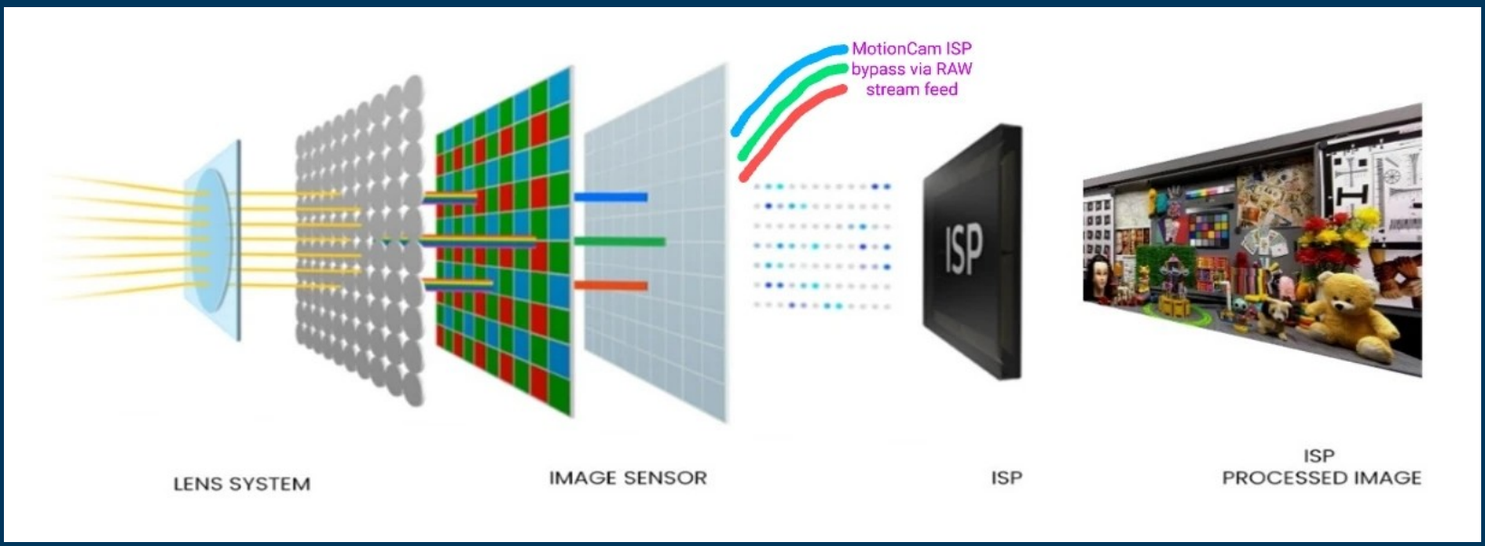
Ready to get into the app at last? I can see you may be still confused, but don't worry! No time to think, we'll make it all make sense! Let's begin!!





Why use the RAW stream (Simplified)

You may be wondering why use the RAW stream for video capture when a dedicated video stream exists. To understand its advantages, consider that all phones capture photos and videos with distinct "personalities." This isn't a flaw; it's a feature. Manufacturers tune their camera processing algorithms (e.g., saturation, noise reduction, sharpening, tone mapping) to produce results at their discretion.



Your phone's Image Signal Processor (ISP) efficiently processes camera data, much like a chef prepares a meal from ingredients. Ideally, this should suffice. However, OEMs often mis-tune these settings, leading to poorly processed results. This isn't a hardware issue; rather, these processing settings are often influenced by consumer taste and demand. Imagine if a chef wanted to cook an amazing grilled chicken, but the restaurant owner forced him to only serve fried chicken since he thought people preferred it better! The chef is simply doing as instructed unfortunately, and he’s not taking your (the client) special orders!

The RAW stream however offers pure, unprocessed sensor data, which is crucial for typical RAW imaging that requires preserving the full frame data. By harnessing this aspect, our unique approach allows our Android app to capture RAW video (every frame of the video can be a full and complete RAW) and bypass the Image Signal Processor, giving us complete command of the data and allowing our own pipeline to take over (which you fully command). Consequently, our capture results when compared to stock camera apps and other third-party solutions may not even appear to be shot from the same device, and can achieve video quality previously thought impossible on phones!

So what's the catch?

Although the benefits of RAW video and the additional capabilities that come from having complete control are undeniable, here's where we arrive at the complexities of this approach.

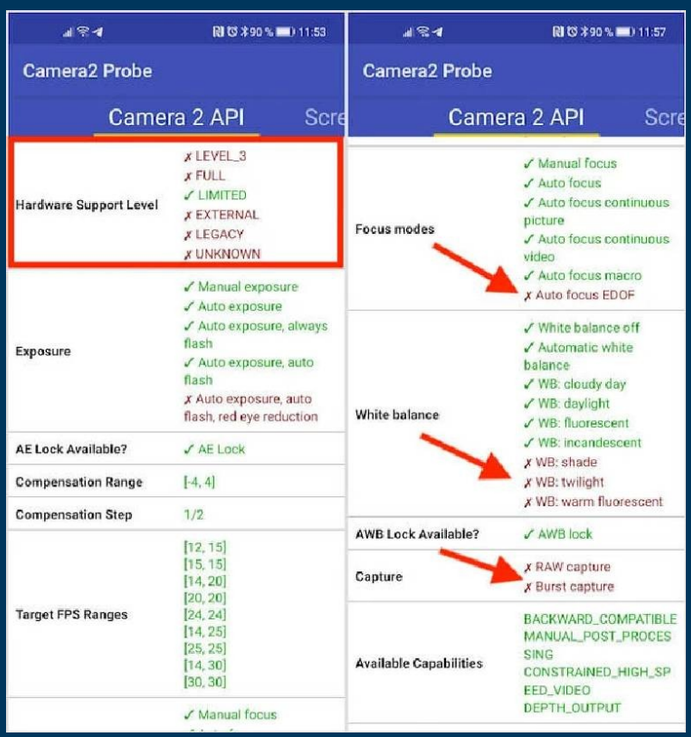
1. Camera2API support

Do you recall those developer tools we mentioned previously to enable control of the cameras and such? The Camera2API is a framework on Android that allows third-party apps to access a device's camera. Manufacturers use this to set what camera settings, such as exposure, ISO, frame rates, and resolution, can be controlled and accessed by developers (amongst many other parameters). You can use a Camera2API tester app to find out in better detail about your device's current support levels and functions shown.

However, a problem arises when manufacturers limit or hide capabilities that are present in their own stock camera apps. For example, most notably they may restrict access to higher frame rates or resolutions as well as In-Sensor Zoom crops. This can be an issue for third-party apps like MotionCam, which requires specific functionalities like RAW stream access. If a manufacturer hasn't enabled the ability to shoot RAW images, or in other words, hasn't exposed the RAW stream, the app simply won't be able to run on that device.

While this is less common on flagship phones, it's still an issue for many lower-end devices.

If you encounter these types of limitations, we strongly encourage you to submit feedback to your device manufacturer and let others in your device communities know. Raising awareness can and has sometimes prompted manufacturers to fix these issues in future firmware updates. Strength in numbers!



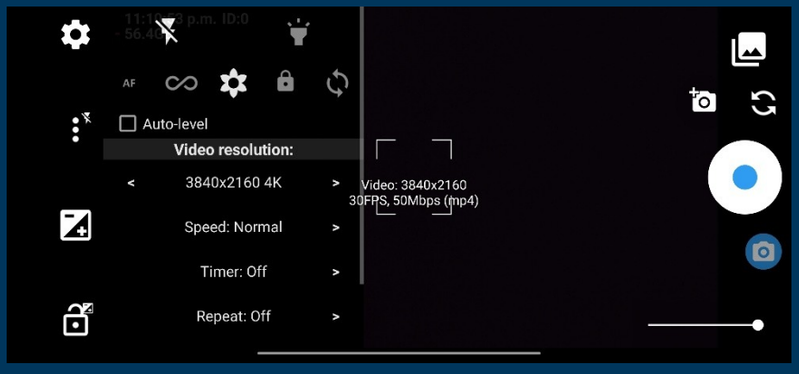
2. RAW stream limitations

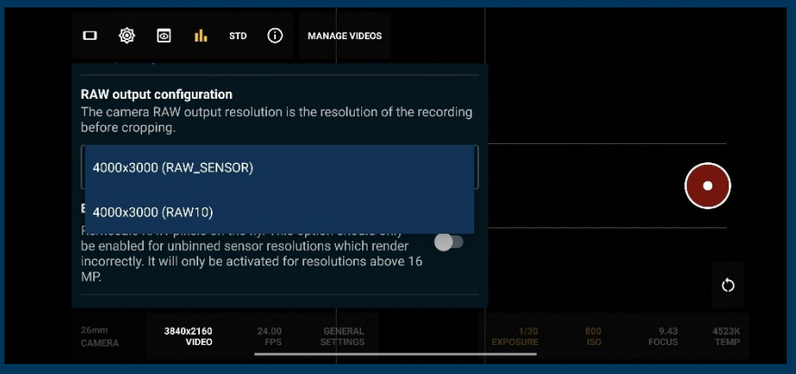
As already stated, the RAW stream provides pure and unprocessed sensor data that the app intercepts before the Image Signal Processor (ISP) touches anything beyond basic sensor operation. The issue however is that the RAW and video stream do not share the same device function access keys.

As an example, if you use a typical video app and request a 4K 60fps SDR video, the device will generally have a capture ‘key’ to use exactly 3840 x 2160 8-bit 60fps HEVC video or something of the like without issues, assuming an acceptable level Camera2API support. In the case of MotionCam however, requesting 60fps means the app is asking your phone for 60 full color depth 12MP images, depending on your sensor, then converting it down to your settings (eg. crop into 4K and reduce down to 8-bit).

What this means is that OEMs may sometimes open up 60fps for video modes, but not for RAW stream capture, leading to a gap in capabilities that shouldn’t exist and is generally a result of poor Camera2API implementation.

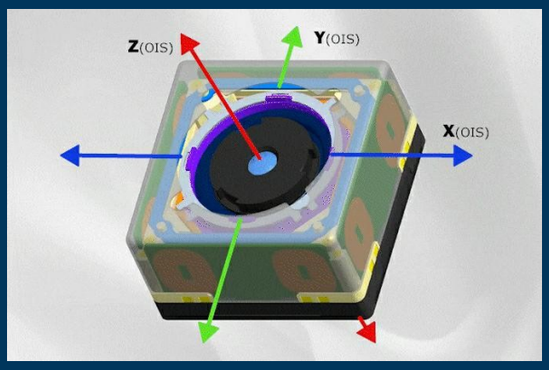
The same goes for resolution settings - where a video stream app may ask for a 7680 x 4320 (8K) processed HEVC video at 30fps, MotionCam would ask for 30 individual, full-color-depth RAW images per second at 50MP.





To be clear, these are strictly firmware limitations at play and will vary per device. Any capabilities available for video mode should be equally available for the RAW stream since all sensor data is initially acquired at the RAW level (even if then processed by the Image Signal Processor afterwards). Google Pixel devices are specially known to show most streams correctly for example, but this is unfortunately the exception rather than the norm.

3. Optical and Electronic Image Stabilization

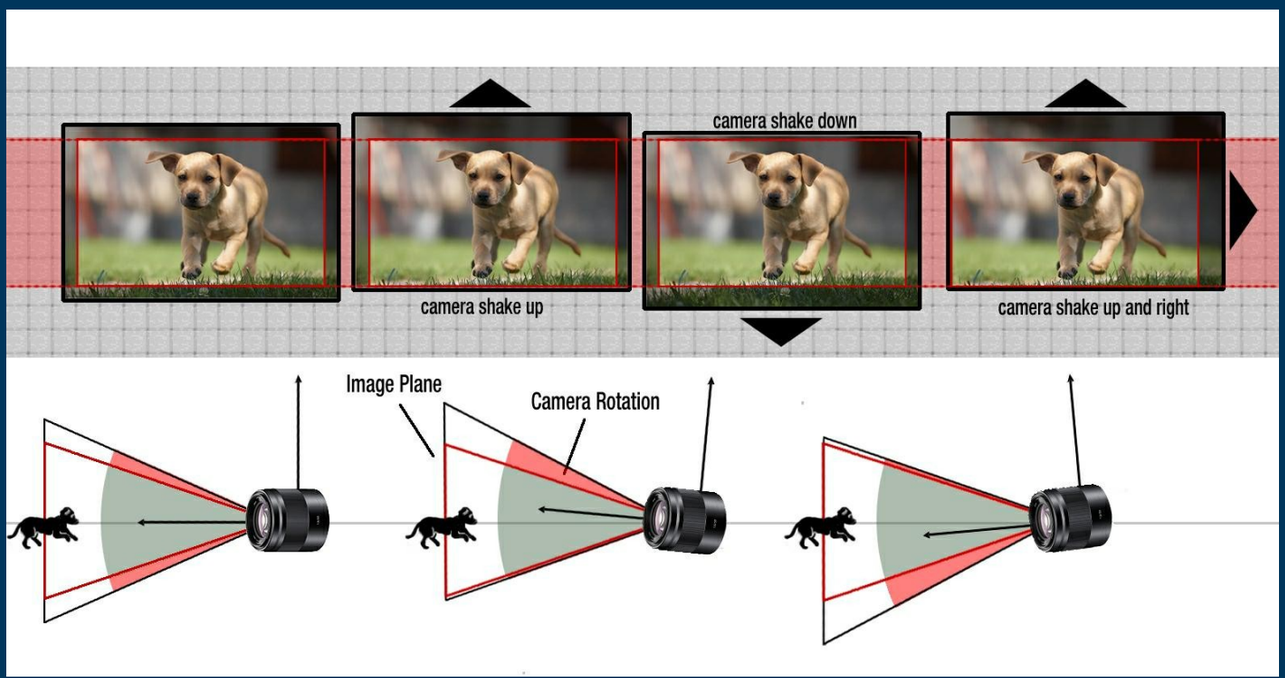


When shooting photos or videos, camera shake is a common issue, especially without a tripod or gimbal. Optical Image Stabilization (OIS), a hardware solution, combats this by physically moving the camera lens to counteract minor shakes and smooth out motion as best as possible, despite lens wiggle room limitations. Optical Image Stabilization (OIS) is generally the first line of defense against camera shake, especially when using a phone or camera without additional external stabilizers. Although it is an excellent tool, it is often insufficient on its own notably during video, bringing us to the next line of defence.

Electronic Image Stabilization (EIS), also known as Digital Image Stabilization, is often used alongside OIS (Optical Image Stabilization) to smooth out shaky video footage. While users might not always realize it, EIS (Electronic Image Stabilization) is crucial for handling the majority of stabilization during significant movement, especially if shooting handheld. In some cases, EIS may even be the only stabilization available on a device!

Why is this important? Well, EIS is a pretty cool trick, but guess where it is coming from? If you guessed the Image Signal Processor (ISP), congratulations! I owe you a virtual cookie compressed into a .rar file, you'll just have to figure out how to un-zip it.

The ISP does a lot of work behind the scenes, including video stabilization. MotionCam bypasses the ISP entirely as mentioned however. Think of it like this: the ISP is the chef, and you're Gordon Ramsey telling it to get the hell out of your kitchen; all the work now falls on you (yes, that includes EIS…)



In fact, some may even think that stabilization is broken in the app altogether —if you are one of those – aha! We meet at last… Now give me that .rar cookie back and go and sit at the corner for calumniating us on that Google Play review. We know what you did..! What's that? You want to give the app 5 stars instead now? Ok, have it back…

Back on track though, this means when you're shooting with MotionCam, you don't get the ISP-sourced stabilization because a RAW stream is, well, RAW. It's pure, unprocessed data. EIS requires the ISP to perform a few key tasks like…

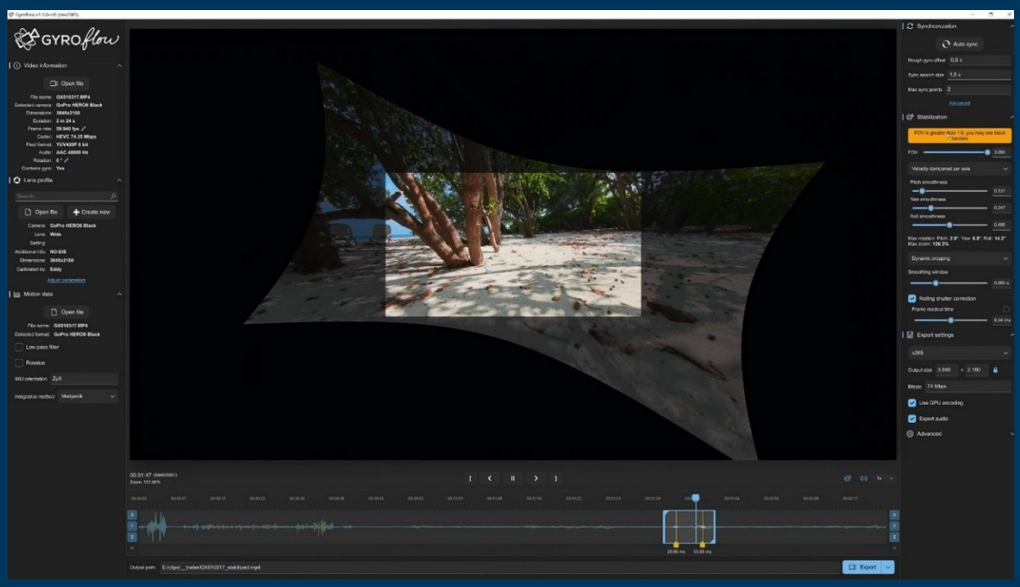
* **Dynamic Cropping**: It crops the video in real-time to smooth out shaky movements.

* ***Rolling Shutter Correction***: It fixes that "jello effect" that can make straight lines look wobbly during movement and panning.

* ***Image De-warping***: It corrects lens distortion to make everything look more proportional (edges in particular).

So, while you might have some shaky footage at first, the trade-off is that you're getting a completely untouched, pure RAW file and the full sensor area to play with. You're in complete control of the final image stabilization, without any of the ISP's automatic (and sometimes heavy-handed) cropping. It's the RAW data versus a "stabilized" convenience, and sometimes, that RAW data is exactly what you want.

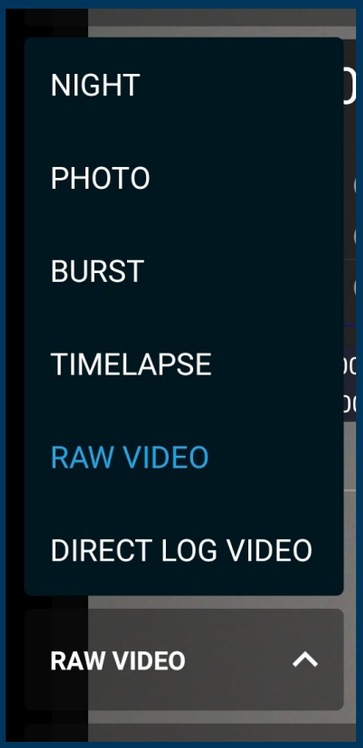
OIS (Optical Image Stabilization) also continues to run smoothly as its lens-based mechanism doesn't conflict with the RAW video stream, nor is it operated by the ISP. For even better stabilization, you have the option to use the EIS/video stabilization tools in any capable editing program like Davinci Resolve or Adobe Premiere to name a few (as well as Gyroflow, an official app supported option –more on that later), which can provide superior control and results compared to the built-in EIS!



GETTING STARTED

App Modes

It's time to put all that prior knowledge into action! I was probably starting to lose you after all that reading, but rest assured it will be worth it! To begin, you will first need to decide what mode you're going to use. As of this current release, here's the options available…



Night: Best for still situations in low light. Overrides exposure set and stacks multiple photos to reduce noise and increase clarity.

Photo: The original MotionCam experience. Captures images continuously in the background to eliminate shutter lag and improve merge quality. Delivers a shutter-lag free, high quality photo.

Burst: Ideal for capturing action. It buffers a specified number of RAW frames (fps selected) from both before and after you press the shutter button, ensuring you can still save that critical moment even if you missed it! Who says you can't go back into the past?

Timelapse: A dedicated mode for shooting RAW timelapses lovers with additional quality of life options vs standard RAW video mode, such as setting interval gaps between frames and total a session duration timer to turn off screen/exit app after completion.

RAW Video: The core and crown jewel of MotionCam. It records RAW video in MCRAW, a highly compressed app proprietary RAW format. It achieves up to 50% data lossless compression over standard uncompressed RAW video. This format saves space and simplifies file management compared to traditional RAW video formats (like CinemaDNG). Heads up! Timelapse mode also captures with MCRAW outp.

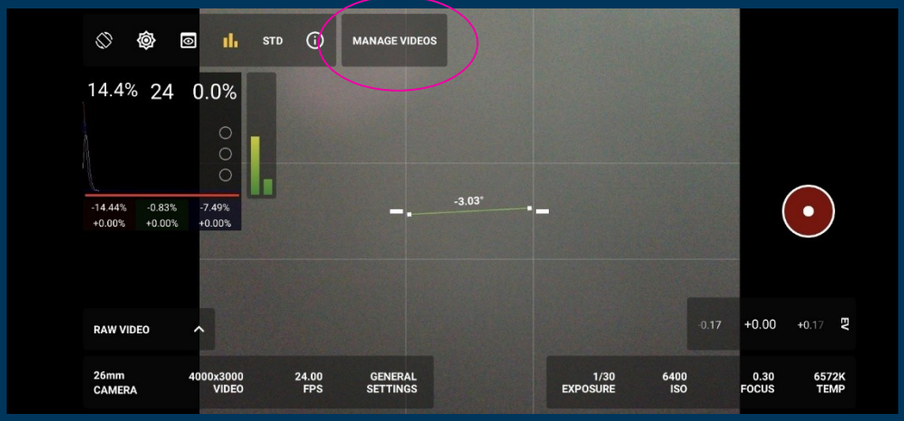
Direct Log Video: Records directly from RAW stream into a compressed video file of your choice instead of MCRAW. Offers a faster, more streamlined workflow by providing a more ready-to-go video file without the RAW size overhead, all while still bypassing the phone's built-in image processor for superior fidelity vs standard video stream options. Significantly more intensive to perform than RAW video as it captures RAW stream data into temporary RAM buffer while encoding it on the fly simultaneously (RAW video mode simply buffers and stores without additional steps). Able to use hardware or software acceleration depending on your device's codecs available, however beware of the performance requirements as this mode will push your CPU and GPU to the maximum!

Shooting your first RAW video

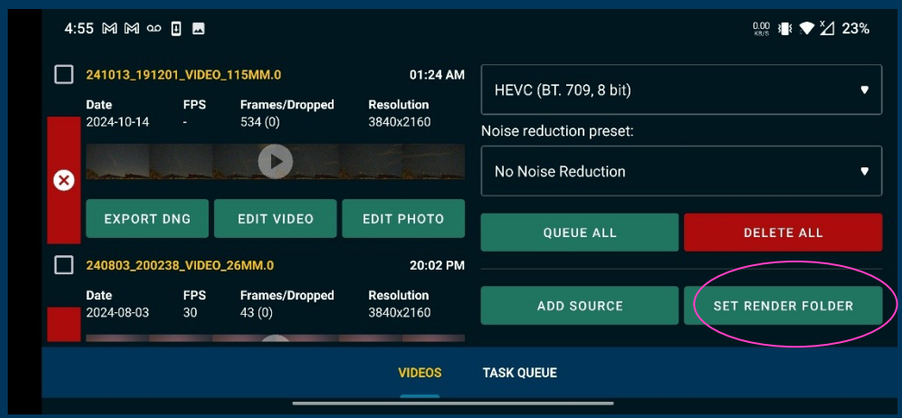
You're probably expecting a long, boring chapter on all the menu buttons. But guess what? We're skipping right to the fun part. The best way to learn is by doing, so let's put that theory into practice. Don't worry about the UI for now; just follow my lead. This is where the magic happens, and it's easier than you think!

RAW video is notoriously complex to edit and manage, but forget about a horrible workflow. This app gives you everything you need to deal with it, all in one spot! And yes, that includes a built in RAW video editor and exporter! Now, let's get our hands dirty. We will assume you've started in RAW video mode, by the way…

1. Start by going into MANAGE VIDEOS

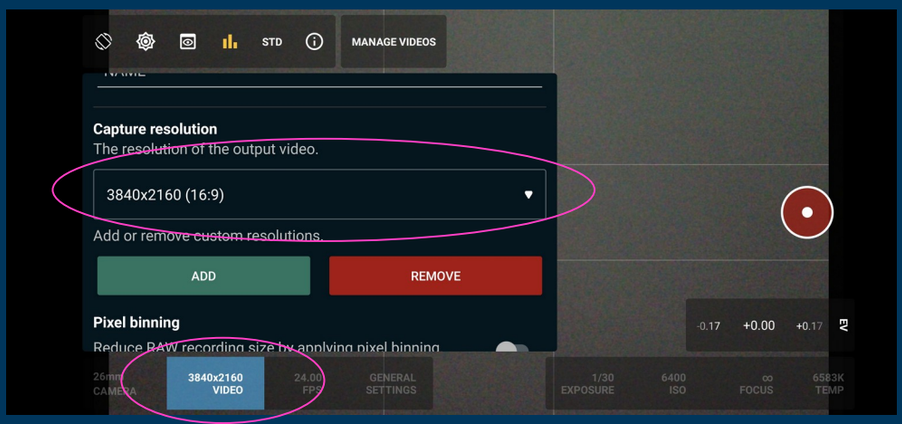


2. Select the SET RENDERER FOLDER button to designate where your RAW video outputs will be exported on completion



After navigating to whichever folder or location you selected, you will be brought back to the MANAGE VIDEOS menu. Simply swipe left gesture to back out of it

3. Once you are brought back, tap on the resolution CAPTURE SETTINGS button, this will open up a sub menu. Simply press on the Capture Resolution box (press ADD to create your own, in case your desired resolution preset is not listed)



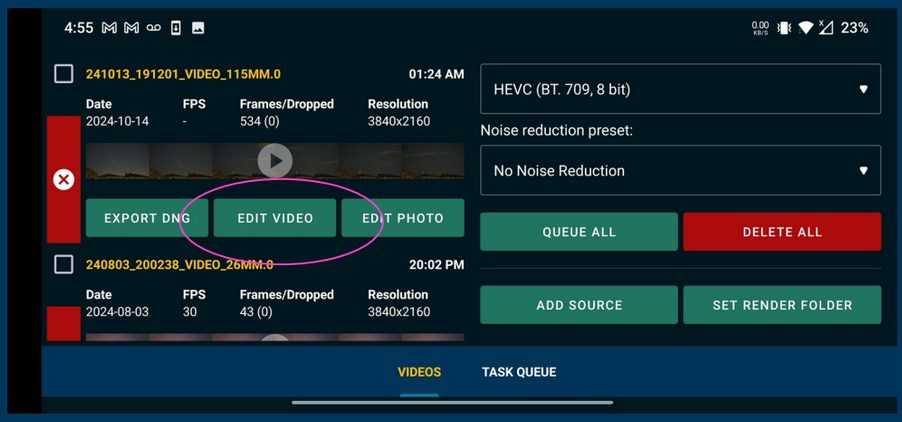
Once selected, press anywhere outside of the menu on the screen to hide it. Now we are ready to shoot!

4. Ready? Simply press the red Capture button to begin recording! Once you do, you will notice the User Interface will change

You now have the option to pause/resume the recording (below capture button) as well as the top left will indicate capture session details! Don't worry too much about those, we will explain them later (the most important one being dropped frame, and storage remaining of course)! The capture parameters will remain functional however once you begin recording you can no longer adjust resolution and other such details, beware!

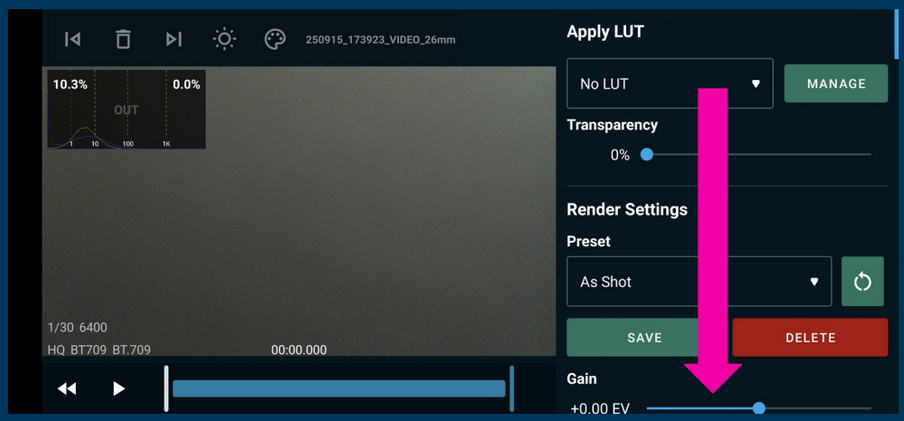
5. Simply press the red Capture button when you are ready. The User Interface will then once again revert back to normal. Press once again on the MANAGE VIDEOS button to re-visit your MCRAW (MotionCam RAW) video captures.

6. You will now see your recent capture (or captures… Did you get carried away by any chance? They pushed crazy numbers on the file size counter, didn't they?? I know, it's beautiful indeed!)



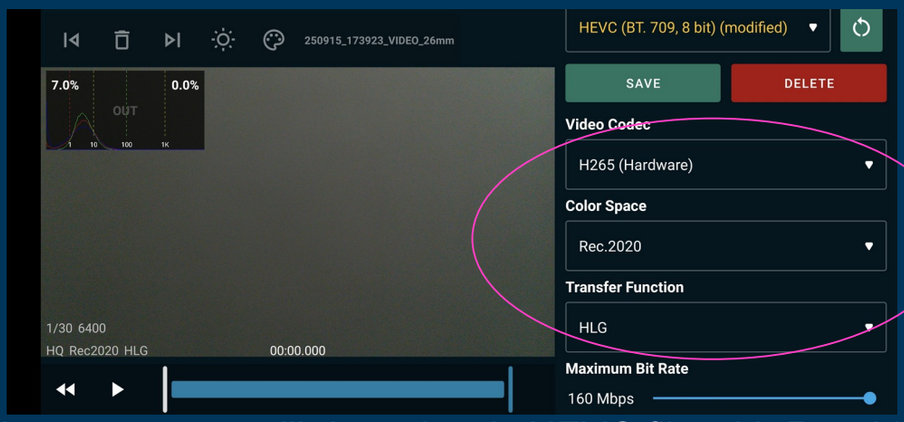
On your desired reel, you can observe three options of interest, EXPORT DNG, EDIT VIDEO and EDIT PHOTO. For this demonstration's purposes, select EDIT VIDEO to continue ahead! (Please don't press EDIT PHOTO yet, you're guaranteed to be sucked away and distracted by the cool features there… Yes, this could be reverse psychology…)

7. Assuming you didn't get sidetracked yet, you will now have opened the EDIT VIDEO menu, which allows you to edit, grade, adjust and denoise your MCRAW files! RAW video editing has never been this easy!



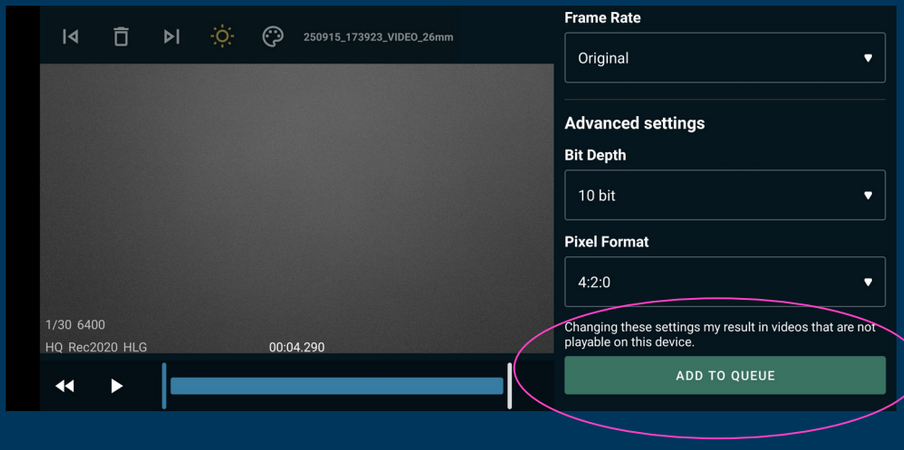
I would heavily encourage you to play and experiment with the sliders to see how you can warp and modify the source content. Do note the preview is low quality however!

8. After adjusting your video to taste (or not, that's also totally fine!) simply scroll towards the bottom of the settings to prepare your export settings.

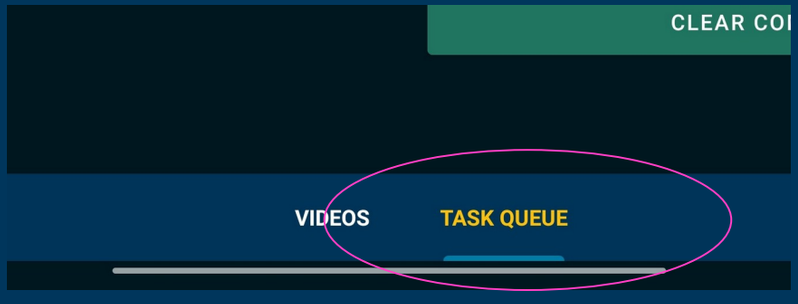


For simplicity purposes, we'll do a classic HEVC file with Rec.2020 in HLG, this is what you may already know as delivery ready HDR video.

9. Scroll down one more time, we can skip the denoising options for now as they can drastically impact exporting time



Simply press ADD TO QUEUE, and this will send you back to the MANAGE VIDEOS menu. You can follow the export process time remaining on a notification that will appear, or on the TASK QUEUE menu.



Congratulations! You've Filmed Your First RAW Video

You've successfully filmed, edited, and exported your first MCRAW! You can find the result in your gallery if your device supports the codec. You will notice a significant

improvement in the overall output quality compared to standard video apps—this is the power of MotionCam's control.

Try the Direct Log Mode for a Faster Workflow

If you are interested in a faster workflow, you can try Direct Log mode. This mode performs all the encoding steps we've just done, except on-the-fly during shooting, so you don't have to use the in-app editor afterward!

Adjustments: Adjustments are made in the Direct Log encoding menu *before* you shoot.

A Note on encoding: Using Direct Log is a **commitment**. The data is "hard baked" and irreversible because it discards much of the sensor data to save a significant amount of space. You will not produce a MCRAW file, and the editor will not work on these files.

Viewing Your Files

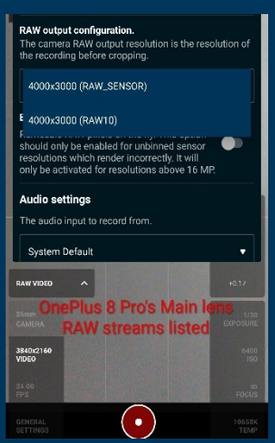
Not all codecs and formats are viewable with default players. To open specialty files like ProRes, APV, and AV1, you can use a dedicated tool like VLC or a video editor.

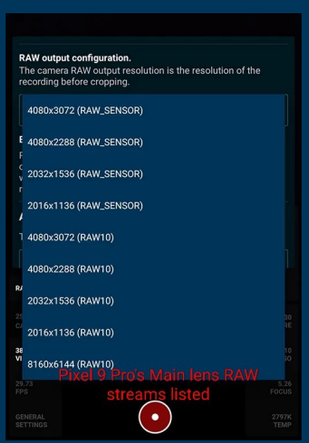
Explore RAW Output Configurations

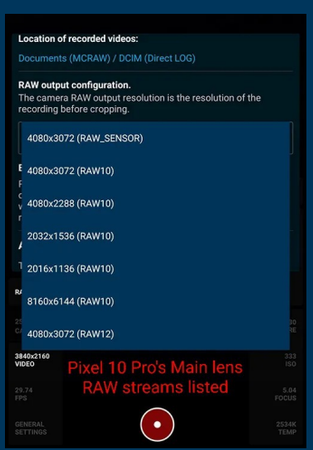
Before we dive into the user interface, it's crucial to understand the **RAW output configuration setting**. You'll find this in the RAW Video, Direct Log Video, and Timelapse modes, below the resolution crop settings (scroll down in sub-menu).

Source Selection: This setting tells the app which RAW stream to use (e.g., "give me 12MP RAWs"). It does not in itself crop your video (cropped streams do exist, however).

Explore & Test: Available streams will vary greatly by device. Explore and test different options. Some may unlock higher frame rates or special features like 12-bit modes







USER INTERFACE & FUNCTIONS

We have arrived at the powerful user interface and inner workings of the app. From this section onwards, I will unfortunately have to be far more technical and specific about the details and functions at play, as it's critical to define what each function does to extract their maximum potential. My recommendation is to go through this section on-demand rather than in a linear fashion (unless of course, you're looking to learn about every literal aspect of the app, that works too!)

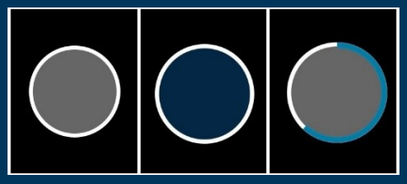
We will now provide comprehensive detailing of what each button, function and tool in the app achieves. We fill first start by covering the Interface of the main modes and their respective overlays during capture. Also covered afterwards will be the sub-menus of the main modes for capture screens.

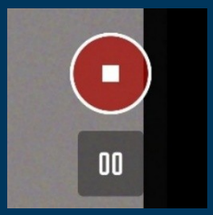
We will then cover the Manage Videos (Video Manager) Interface as well as we will also explain and detail functionalities of the Photo, Video and DNG renderers.

Lastly, we'll cover some important functionality aspects as well as terminology. For additional information on further workflows and handling of exported outputs, skip to the next section ahead (Usage and Workflows).

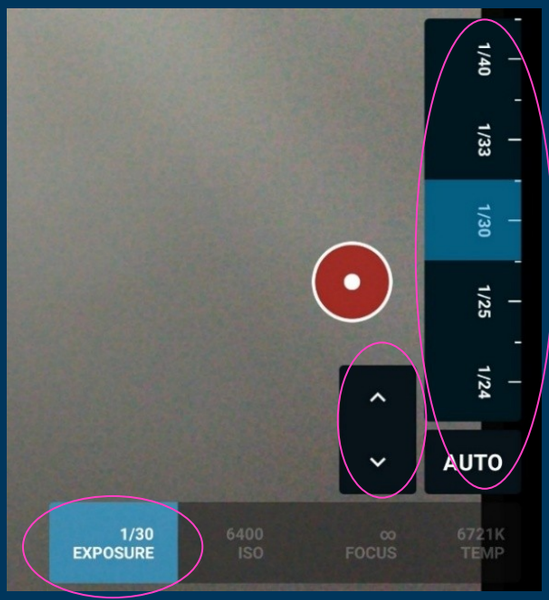
Capture/Record Button: Tap to Capture with universal function across all modes. Starts/Ends recording sessions during Video modes. Press and Hold to initiate Buffer Recording mode (captures video only within RAM buffer space to avoid storage bottlenecks, jump to page ## for more information on Buffer Recording).

A Red button indicates a video mode. A Gray button indicates photo mode (will flash blue when pressed in Photo modes to indicate successful capture and will fill blue circle during night to indicate ongoing exposures



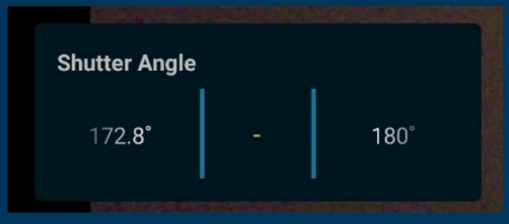


Stop Recording + Pause/Resume Button: A Square inside red circle indicates an ongoing recording session. An icon nearby showing ⏸ / ▶ will appear for video modes during active sessions. Press to Pause or Resume the recording sequence for cuts without stopping the session. Press the Record button to end recording.

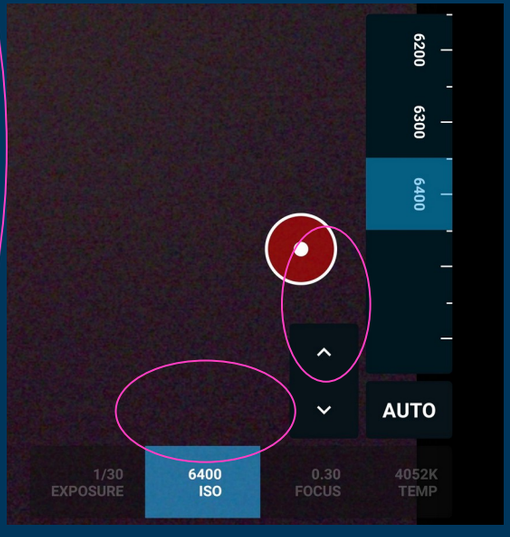


Exposure/Shutter Speed: Shows the current Exposure Length/Shutter Speed in use. A white value indicates AUTO mode is active and the Camera2API automatic exposure driver is in control. Tap to open/close manual control interface. Holding the Exposure value will turn it yellow and manually lock it. Be advised, as of this current version, there is no manual Exposure Priority mode available, full auto or full manual only therefore.

Scroll up/down on slider values to manually adjust Exposure with linear precision. Tap up/down on the arrows to increase/decrease Exposure in fixed values. AUTO button restores Exposure selection back to automatic. Note, if the exposure value turns red, it indicates that your currently selected shutter speed has gone below the framerate you've selected (will shoot below target FPS)



Shutter Angle: Upon pressing on the Exposure value, a special control will appear on the bottom left side. Slide left and right to change Shutter Angle relative to current framerate selected. A "-" value indicates the mode is not engaged. Press AUTO to any selection

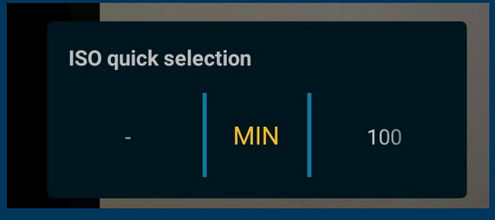


ISO/Sensor Gain: Shows the current ISO in use. A white value indicates AUTO mode is active and the Camera2API automatic exposure driver is in control. Tap to open/close manual control interface.

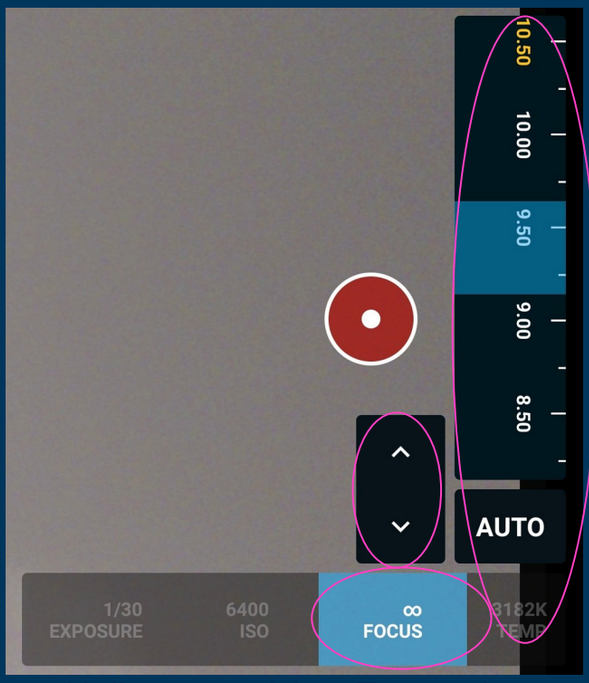
Holding the ISO value will turn it yellow and manually lock it. Be advised, as of this current version, there is no manual Exposure Priority mode available, full auto or full manual only therefore.

Scroll up/down on slider values to manually adjust ISO with linear precision. Tap up/down on the arrows to increase/decrease ISO in +/- 100 increments. AUTO button restores ISO selection back to automatic

ISO Quick Selection: special control will appear to change between ISO value indicates the mode is any selection.



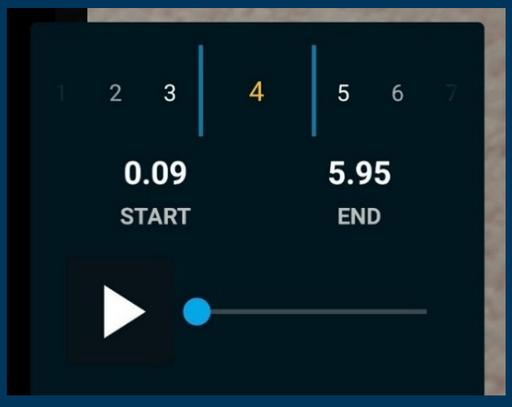
Upon pressing on the ISO value, a on the bottom left side. Slide left/right presets in +/- 1 stop increments. A "-" not engaged. Press AUTO to undo



Focus Ratio/Distance: Shows the current focus ratio, as defined by Camera2API. A lower value indicates farther focus distance and a higher value indicates closer focusing distance. A white value indicates AUTO mode is active and the Camera2API autofocus driver is in control. Tap to open/close manual control interface.

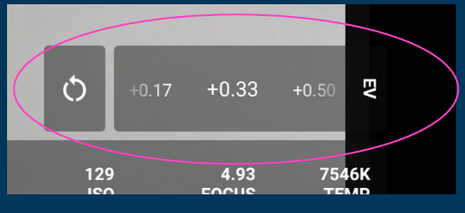
Holding the Focus value will turn it yellow and manually lock it. A figure eight "∞" value indicates infinity focus range, however exercise caution as it may not always be the correct infinity calibration on every device. Yellow values on slider rack indicates extended focus ranges that may not be supported (out of spec), however may provide closer focusing capabilities than reported by Camera2API parameters.

Scroll up/down on slider values to manually adjust Focus ratio with linear precision. Tap up/down on the arrows to increase/decrease ratio in +/- 0.01 increments. AUTO button restores Focus back to automatic. Ratios are measured in diopters (1/meters = ratio), may be uncalibrated however.



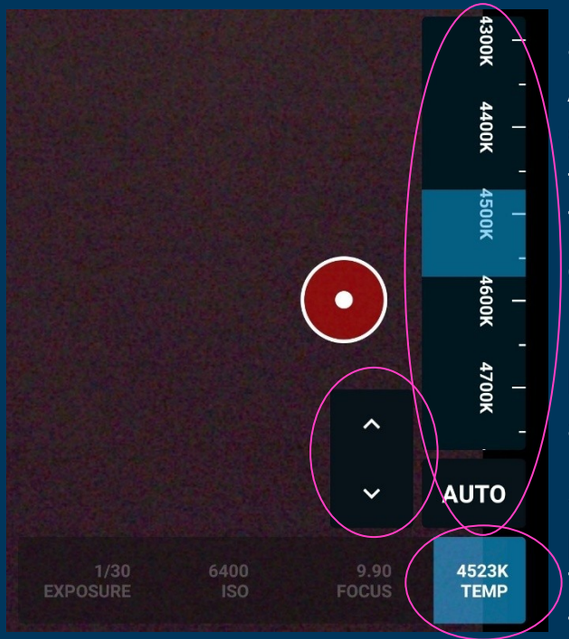
Focusing Rack Controls: Upon pressing on the Focus value a special control will appear on the bottom left side providing focus racking controls. To set ranges of focus rack, first set a manual focus range. Long press on START or END values and the current manual Focus value will be defined accordingly. The top number indicates the duration (seconds) of the focus rack change. Press ▶ / ⏸ icon to begin/pause rack.

The blue dot will traverse the line to indicate progress of rack session (left side = START, right side = END). You may also hold the blue dot and manually swipe it left/right to change focus within the defined START and END ranges only as an alternative to the manual Focus slider controls.

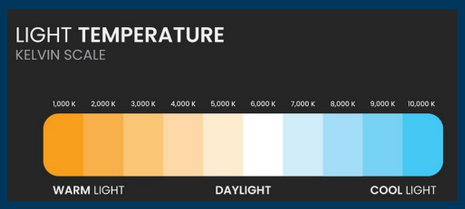


Exposure Compensation: Sets the Exposure Value Compensation of the Auto exposure driver. Controls both ISO and Exposure/Shutter Speed to achieve desired Exposure offset. A +0.00 value corresponds to Off state. Slide numbers left/right to increase or decrease compensation in small increments.

Every +/- 1.00 change corresponds to a Full Exposure Stop (1 stop = doubling/halving of light intake). Press circular arrow icon, or return to +0.00 to undo or exit mode. Using manual exposure settings will null current selection



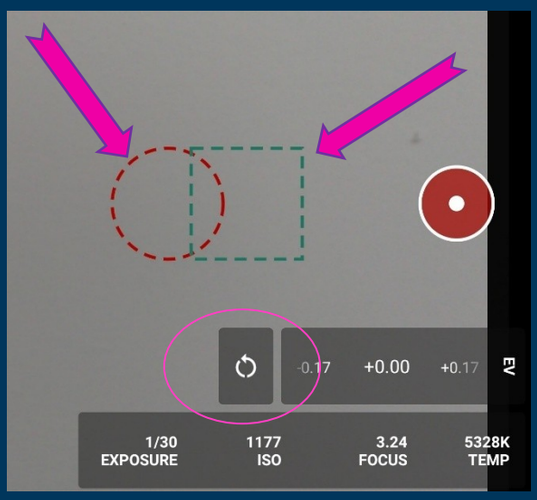
Color Temperature: Shows the current color temperature applied in Kelvin temperature scale. A white value indicates AUTO mode is active and the Camera2API automatic color temperature driver is in control. Tap to open/close manual control interface.



Holding the Temperature value will turn it yellow and lock it. Can be locked independently of exposure controls (doesn't impact automatic exposure driver).

Scroll up/down on slider to select Temperature with linear precision. Tap up/down on the arrows to increase/decrease Temperature in +/- 100 Kelvin increments. AUTO button restores Temperature selection back to automatic

Does not impact MCRAW or DNG quality as only applied to EXIF data (the attached suggested setting) of RAW formats, however gets baked-in after encoding or compression into Photos, Direct Log videos or exported codec outputs.



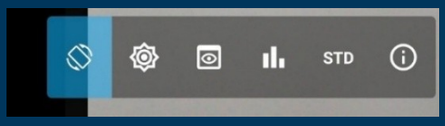
Spot Metering/Focusing Indicator: When you tap anywhere on the viewfinder, the app displays a circle for spot focusing and a square for spot metering.

The circle will prioritize the autofocus driver area, while the square will prioritize the auto exposure driver for the area it covers as well.

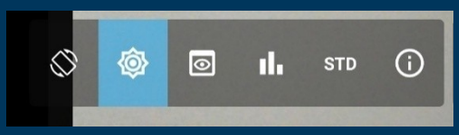
A red value indicates that the settings are still being adjusted, but a green value means the app has successfully locked on to the target region. Changes within the Square/Circle areas may result in loss of lock-on, so color will switch back to red while the settings hunt again. You can lock any acquired settings by pressing and holding the Exposure, ISO, or Focus values on the settings bar, and you can also drag the circle and square independently to different areas each.

Any changes via manual modes (except on Temperature) will exit respective Spot mode individually (Exposure/ISO change = Loss of Spot Metering and Focus change = Loss of Spot Focusing).

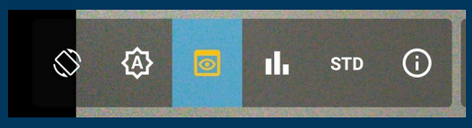
To exit Spot modes completely, simply press on circle arrow icon to undo both



Display Rotation/Orientation: Allows for control of the display's current rotation preferences. Tap to alternate between Portrait (Sideway rectangle) and Landscape (Upward rectangle). Press and Hold to unlock rotation (follows system orientation preferences). A diagonal rectangle with arrows indicates unlocked rotation is active.



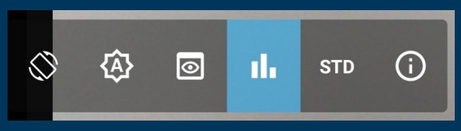
Display Brightness: Provides control of the display brightness preferences. Tap to alternate between Full brightness (Indicated by white filled sun) or System/Phone setting (Indicated by Sun with letter A inside)



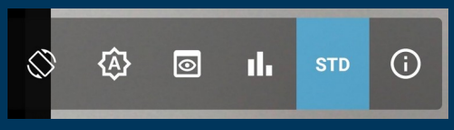
Direct Preview Mode: Enables/Disables Direct Preview Mode (DP Mode). Direct Preview allows you to see your current Capture settings real-time and bypasses Android viewfinder which can provide inaccurate scene previews that show incorrect image and is subject to ISP processing unlike the captured outputs. A yellow state indicates On, and a white state indicates Off.

Can also be enabled also via hardware button. Press and hold Vol + button to activate. Subsequent Vol + presses will then alternate between Direct Preview, Sensor Clipping and False Color overlays. You may also press Vol - button to exit.

For more information on how to operate Direct Preview Mode, jump to page ## (How to use Direct Preview Mode)



Histogram Quick Toggle: Provides quick toggle to Show/Hide the RAW Histogram (Appears below) from user interface. Tap to toggle. White = Off (Hide Histogram), Yellow = On (Show Histogram)

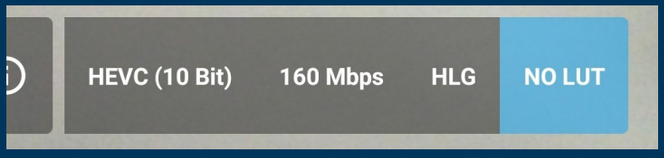


Standard/Maximum Preview Resolution Toggle: Provides ability to quickly change viewfinder resolution from current selection to maximum resolution available. Does not impact capture quality however may impact performance. White STD letters indicate selected viewfinder resolution is active, red MAX letters show Maximum viewfinder resolution has been enabled (jump to page ## for more in viewfinder reason and to enable this toggle as otherwise it will not appear on the bar)

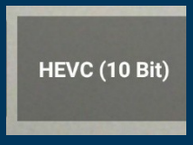
About Us: Opens screen showing App details including Contact Us and Support information, Community groups, as well as additional resources for app usage.

Manage Videos (MCRAW Manager): Appears while using RAW VIDEO, TIMELAPSE and BURST modes. Tap to open the app's built-in MCRAW handler (MotionCam RAW video files).

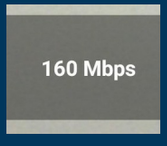
For more info on using the MCRAW Manager, jump to page ##.



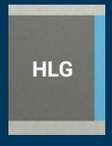
Direct Log Video Settings Bar: Appears only during Direct Log Video mode. Shows settings bar with brief summary of the current encoder settings to be used during Direct Log Video capture.



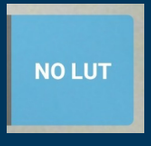
Codec (Color Depth): Indicates the current encoding codec selection and its color depth inside parenthesis.



Bitrate: Indicates the current bitrate or encoder quality level selected. Shows numerical Mbps value or letter name depending on codec (Eg. Will show "**LT**" for ProRes LT encoding grade).

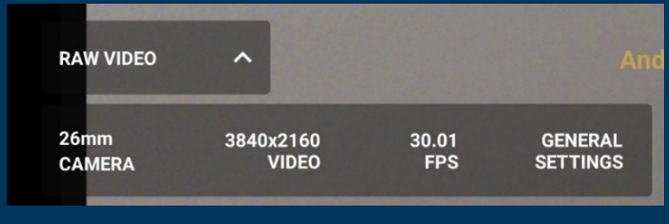


Transfer Function: Indicates the current Transfer Function in use (Eg. HLG, PQ, Apple Log, Davinci, etc.) Does not show Color Space however.

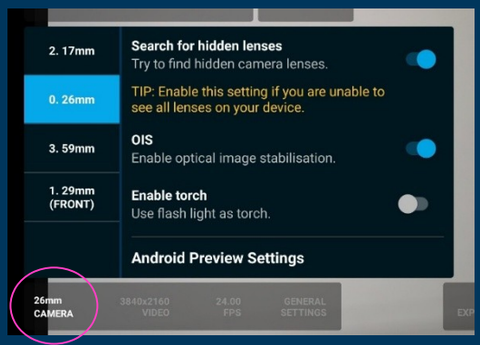


LUT Applied: Indicates the current LUT selection to be baked-in into Direct Log Video outputs. NO LUT indicates no active LUT is in use.

Tap on any of these settings to open the Direct Log settings sub-menu. Press and hold LUT indicator to quickly enable/disable any recent LUT applied. Jump to page ## for the complete Direct Log Video Settings sub-menu details.



Camera & Display Controls: Control interface bar with options to adjust parameters such as Camera/Lens ID in use, framerates, RAW stream and cropping, as well as viewfinder aids. Swipe left/right on Camera lens category to quickly change between camera in use.



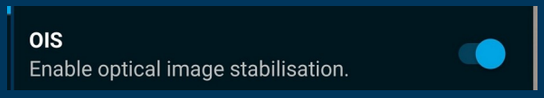
Lens & Viewfinder: Tap to open menu for controls of Lens (Camera ID) in use, adjust additional capture aspects, as well as modify viewfinder parameters.

Camera Lens Selection: Shows the lenses listed as available by the device. Not all lenses shown may work. A blue rectangle indicates the active lens currently selected and in-use. Tap on any lens to switch to it. Selfie lenses will be indicated by (FRONT).

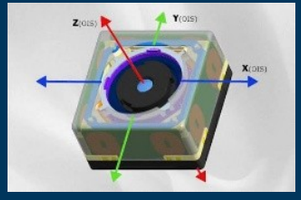
Lenses appear In 35mm equivalent focal length (eg. 24mm = x1 zoom). Please note that the focal length shown may not be exact and is reported as per Camera2API information.

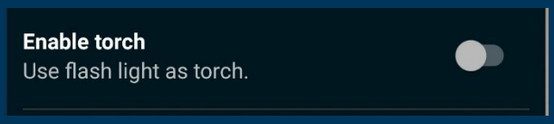
An individual # value indicates a physical ID, and a #/# indicates a logical ID (Eg “2. 26mm” would indicate Hardware ID 2, and “0/2. 26mm” would indicate a Logical ID 0 using physical lens 2). Although uncommon, sometimes duplicate IDs may provide access to different framerates or sensor modes than their main ID, so we encourage you to explore them at least once.

Search for Hidden Lenses: Enable this setting to force a scan for lenses that the device is not reporting. If you’ve ever seen a horror movie, it’s like when a character asks “who’s there?” before getting killed, except that in this case some Camera IDs may sometimes reply like “Oh, you got me!” then reveal themselves. This setting can sometimes find hidden lenses that should appear but don’t. Always try to use it in the event you can’t find a Camera module, however be mindful it can also find broken or duplicate IDs (it’s also not guaranteed to work as OEMs may fully block Auxiliary lens access in some cases)

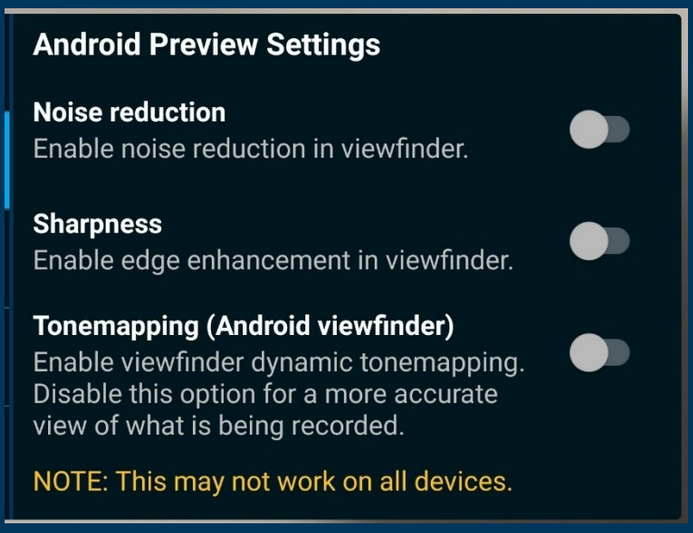


OIS (Optical Image Stabilization): Toggle to Enable/Disable Optical Image Stabilization in supported lenses. Moves camera eye module as anti-shake mechanism. Helps photo and video stability, however should remain off if you plan to use Gyroflow as it will interfere with it. Can potentially hamper gimbal performance as well so use with care.





Enable Torch/Flashlight: Toggle to turn phone torch (flashlight) on and off in Fill mode to provide additional video illumination. Note: may possibly not work with all lenses.

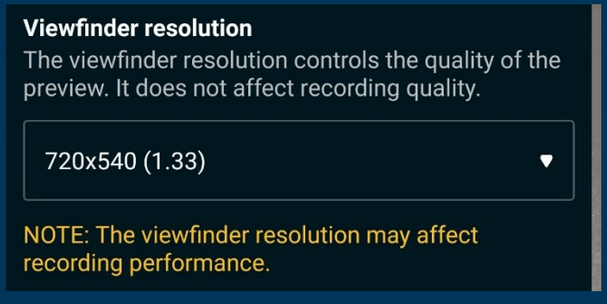


Android Preview Settings: Allows for adjustments to the Android Preview image quality (Jump to page ## for more information). These settings do not impact capture quality.

Noise Reduction: Toggle off to reduce/disable viewfinder noise reduction

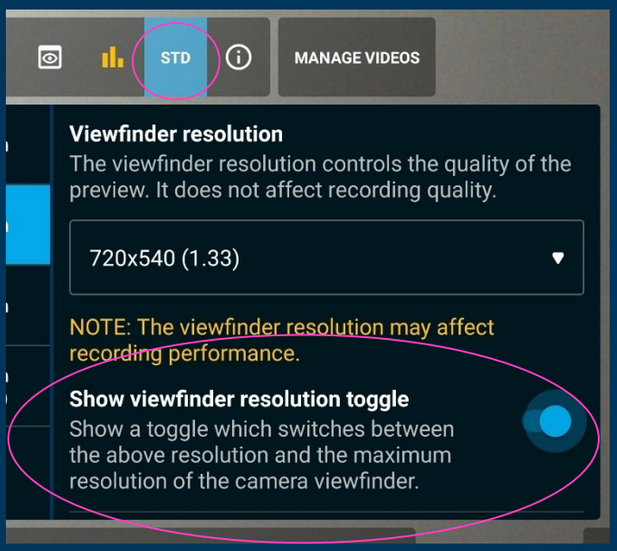
Sharpness: Toggle off to disable viewfinder Sharpening (Edge Enhancement)

Tonemapping (Android viewfinder): Toggle off to disable preview Dynamic Tonemapping that may show drastically different light and color reproduction compared to capture quality. Having this option disabled is desirable for more accurate previewing, however do note it can sometimes cause visual glitches or malfunctions in some devices and may not work (preview freezing, screen halves showing differently, broken viewfinder images, and/or app crashes).



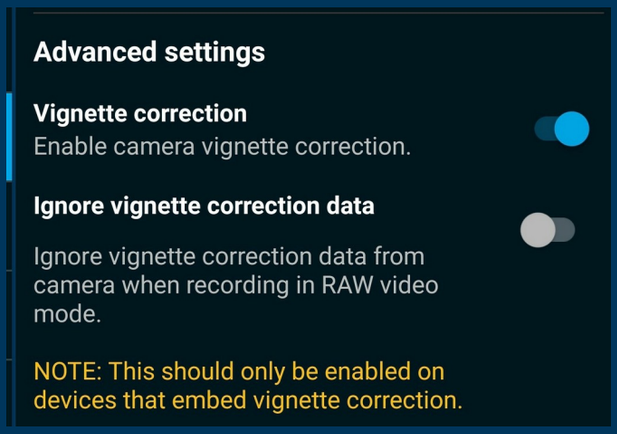
Viewfinder Resolution: Provides control of the resolution rendered for the viewfinder preview. Does not impact capture quality however will impact capture performance (a 480p preview is far easier on the CPU vs a 1080p/4K UHD preview which may heavily tax the system performance.

Note: the ratio of the viewfinder will be shown as a number in parenthesis, in the above example 540 x 1.33 = 720 as an example. Do note that if the ratio chosen can't fit the current capture resolution correctly, you may possibly warp the viewfinder or no longer be able to see the outermost edges on viewfinder, however this will not impact capture quality.



Show Viewfinder Resolution Toggle: When enabled, will show an option on the top quick settings bar to quickly switch between the current *Viewfinder Resolution* (STD) to the Maximum resolution (MAX) listed by Camera2API.

Be mindful of using MAX viewfinder resolution, although it will not impact capture quality, it will degrade device performance due to rendering a high amount of details for the preview. Best used to quickly hone into a perfect focus for tricky scenes with a lot of details. Usage of this mode during captures is not recommended.



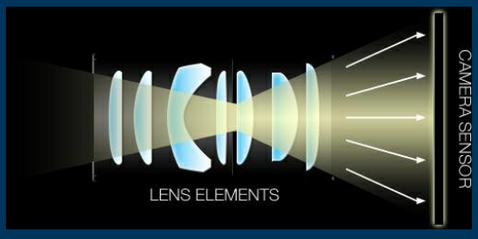
Advanced Settings: Below this area, additional controls that will affect capture quality and lens list are shown.

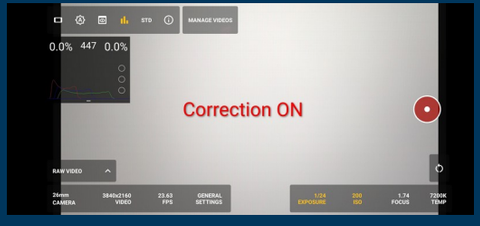
This area should not be modified unless you are having issues or you have duplicate/hidden lenses that may either need to be added manually or removed if they're broken/inoperable

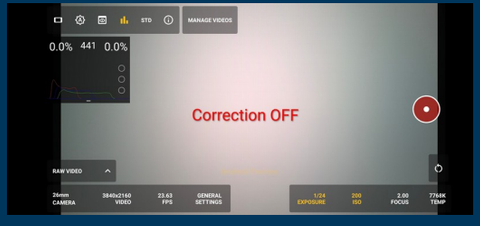
Vignette Correction: This setting allows you to control the digital correction of the natural vignette that all lenses produce. Toggle On to enable and Off to disable.

Vignetting is a gradual darkening twards the edges of an image, caused by the physics of a circular lens where the corners of the sensor receive less light.

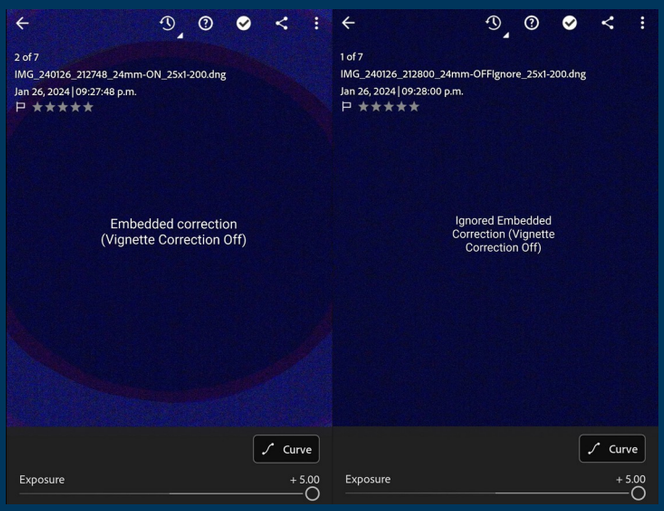
Toggling this setting On applies a digital offset to correct this difference by brightening the corners with a counter gradient. However, be aware that this may also amplify noise in those areas, and in low light, it could introduce a noticeable purple circular pattern resulting from lifting overly dark areas. Note: In rare instances, lenses can exhibit a center pinkish hue with VC



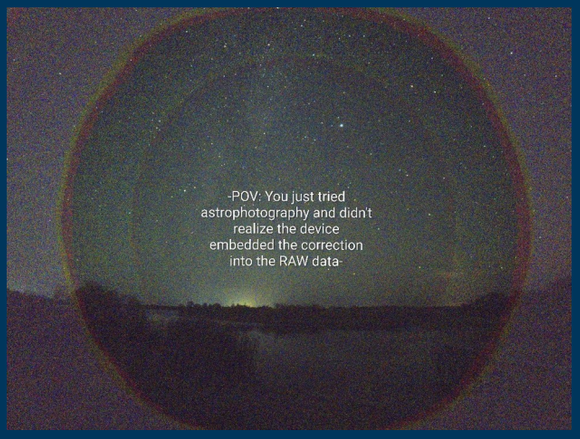




Ignore Vignette Correction Dato Correction: This setting should only be used as a reactive measure, but it's important to understand why. Toggling it on will completely ignore any vignette correction data that the device may generate during capture. Normal correction works by providing an additional bundle of data with the correction offset info in the RAW file exif. This is like when you get fries at McDonald's with a packet of salt—you have the option to use it or not.



The problem is, some devices bake this correction right into the file and don’t give you a choice (I’m looking at you, Samsung and Sony, as a personally traumatized user). It’s like if they added a ton of salt on your fries before you even got them. While this might sound convenient, it means you're stuck with irreversibly altered data that can have more noise, artifacts (Square looking patches, altered colors in borders of lens), and even purple rings due to the device's sometimes-poor correction.



Another side effect is that embedded data + Correction On can double correct the vignetting, further aggravating anomalies. It’s like adding a SECOND pack of salt to an already salty abomination, only a true brine enthusiast would enjoy it.

Using this option purges the embedded correction before the device has a chance to bake it in, but it also purges the app's own correction data, which is usually a bad idea on normally functioning devices. So, if you're getting weird artifacts, try this option. Otherwise, steer clear.

A NOTE FOR ADVANCED USERS: Vignette Correction contains corrections as well for more than just the gradient mentioned. We will touch more on this in the Advanced section; due to optical issues with the IR filter elements on some devices, a disabled vignette can produce a radial color shift that looks like a red patch whenever VC is off. More on this will be covered on page ## to better explain it and provide solutions, however newer devices are less affected by this.

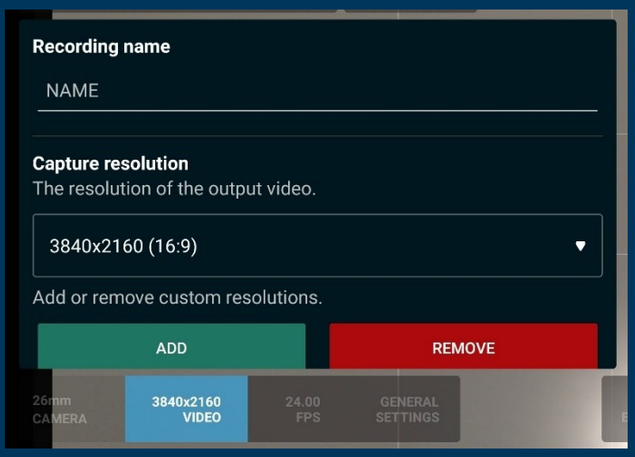
Camera Lens Visibility: This section displays the complete list of lenses available to the app, which are shown on the left side of the menu under Camera Lens Selection. The list you see will vary widely per device, as some camera IDs are duplicates, while others are Logical IDs or Hardware IDs. We encourage you to experiment and confirm which work and which don't. Toggling the Search For Hidden Lenses (Page 37) option will generally reveal additional IDs, but their functionality is never guaranteed.

A blue checked box indicates the lens will be shown in the selection to the left. Tap the box to uncheck then and hide them from the list, or vice versa (after testing them, hide the broken IDs, only show the good or useful ones!)

The reason for this confusing list lies in the different ID types. Hardware IDs are the easiest to understand: they are the actual physical lens itself. For instance, if you have a Main, Telephoto, and Ultrawide lens, their Hardware IDs would be 2, 3, and 4, respectively. However, OEMs can arbitrarily name these, so take this only as a conceptual example.

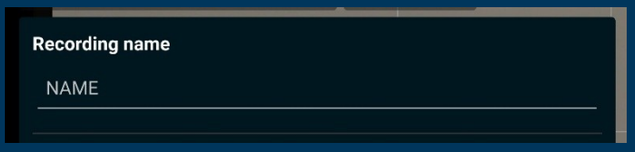
The complexity often comes from Logical IDs, which are denoted by a #/# designation. A Logical ID is a virtual parent ID that acts as an interface. When you ask a Logical ID (say, ID 0) to use a specific zoom length, the phone selects the most suitable underlying lens for you. For example, if you have three IDs, using a Logical ID 0 means you are summoning them all on demand; asking it to use the telephoto means you are using ID 0/3. While you don’t need to know this for MotionCam (the app forces them, regardless), understanding this gives you the reasoning behind the madness.

Finally, some OEMs introduce Sensor Mode IDs, which is a form of Camera2API malpractice. This happens when manufacturers go full IQ200 and map specific functions to their own hidden ID. This is a worst-case scenario, as it means a Hardware ID 2 might have a hidden 60fps mode or RAW access only available on a separate duplicate ID 11. You are forced to switch to that "fake" ID to access the function. We’re looking at you, Samsung…

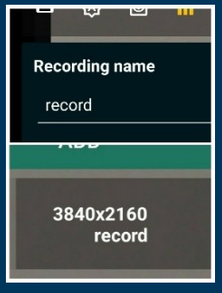


Capture Configuration: Shows the current Capture Resolution (Crop) and the name of your video file (Defaults as VIDEO unless a name is added). Appears at the bottom left settings bar showing the Capture and Display settings. Tap on it to open the Capture Configuration Menu.

Here, you can adjust multiple settings including the famous RAW streams we talked about plenty, as well as you can select the capture crop, set file paths as well as mkre options involving footage capture.



Recording Name: Use this setting to add a custom file extension name to your captures. Simply press NAME to open the keyboard and type your desired name, which will persist until manually removed. This custom name will also be reflected on the bottom-left bar below the resolution and applies to both RAW and Direct Log outputs.



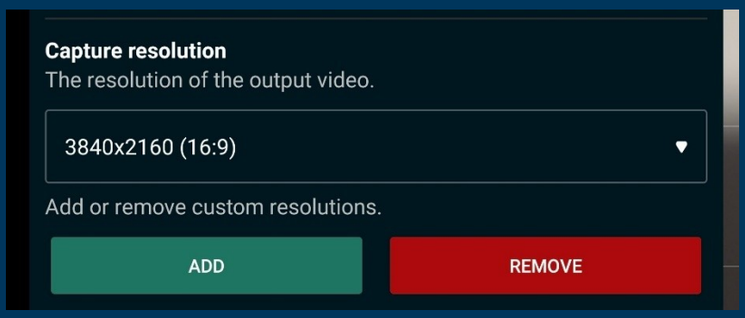
The app's file name is constructed using a clear nomenclature: it includes the Year (YY), Month (MM), Date (DD), followed by the Hour (HH), Minutes (MM), and Seconds (SS). This timestamp is then followed by your custom file name, and finally, the Focal length of the lens (#mm). For Direct Log mode, the app automatically adds two extra extensions to the file name to specify the Transfer Function (e.g., Samsung Log) and Color Space (e.g., Rec.2020).

See the examples below for how the files would appear relative to the mode in use…

MCRAW video: YYMMDD_HHMMSS_NAME_#mm.mcraw

Direct Log Video: YYMMDD_HHMMSS_NAME_#mm_Transfer Function_Color Space.format

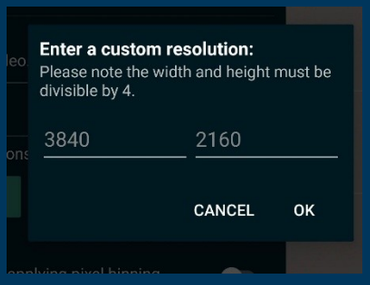
Note: the .mcraw files will not show on filter explorers if under private storage and must be first transferred outside of it. More on page ##.



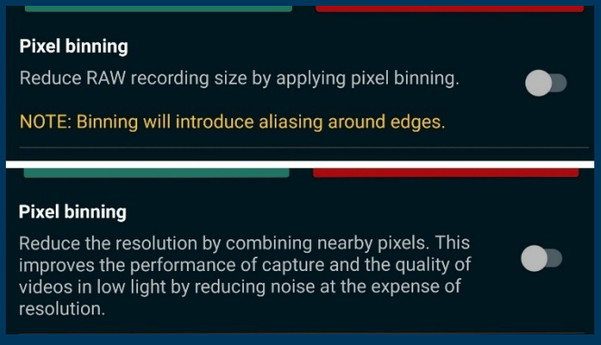
Capture Resolution: Use this setting to define the crop resolution for the current sensor stream, which is distinct from the overall Sensor Mode or RAW stream selection.

This crop essentially tells the app what final resolution you want to extract from the available stream. For instance, from a 4000x3000 stream, a setting of 3840x2160 (16:9) will discard all pixels outside that range.

Be aware that a higher resolution demands more device performance, and resolutions exceeding 3840x2160 can sometimes interfere with 10-bit HEVC encoders when rendering compressed video (refer to the FAQs for details).



To use a different crop preset, press the rectangle box showing the current one. You can also create your own by pressing ADD or delete custom ones by selecting and pressing REMOVE.



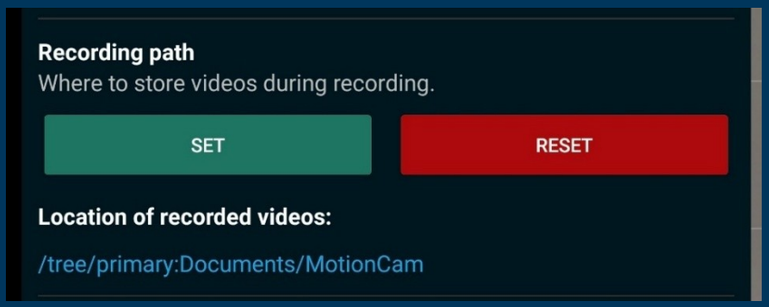
Pixel Binning: Enables resolution binning, which reduces the resolution of the currently selected capture resolution by a 4:1 ratio without cropping (eg. turning a 4K UHD selection into 1080 FHD). You should use this option primarily to save storage space and reduce the performance tax on your device.

The app uses the Capture & Display Settings button as a visual indicator: a yellow value means binning is active, while a white value means no binning is in use. It's crucial to note that binning works differently depending on the mode:

RAW Video: Binning primarily to save space but does not resize the data. This can introduce some aliasing (noted around edges) and impacts fine details, but it drastically reduces file size.

Direct Log Video: The app instead uses super sampling during encoding, meaning that as an additional benefit, at the cost of resolution, you get an improvement in low-light performance rather than just saving space.

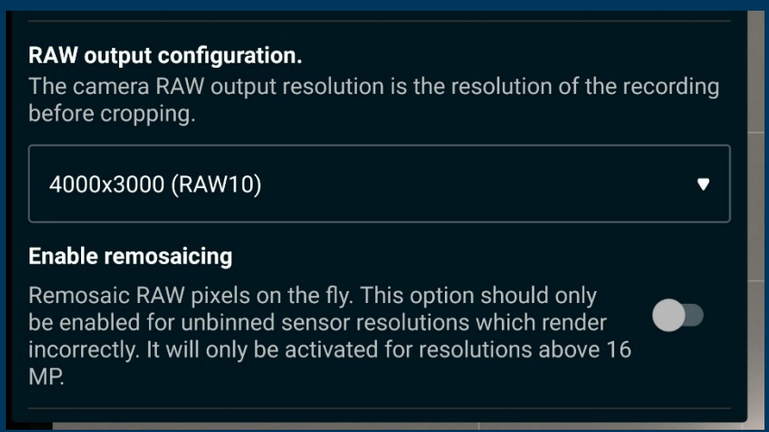
The description for Pixel Binning will change depending on the mode in use to advise of this.



Recording Path: This option will allow you to designate the location or folder on your device to use for saving your captured outputs and/or designate External Storage locations like SSDs and such.

Press SET in order to open the Android file explorer. Any selected location will be used afterwards to store captures. This option can be undone by pressing the RESET Button.

Any active location in use will be displayed on the bottom part in blue (Location of recorded videos:) and you can observe the file path chosen. The default location unless set otherwise will be Internal Storage/Documents/MotionCam



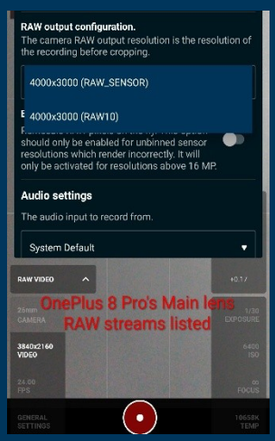
RAW Output Configuration: This is one of the most critical functions to understand in MotionCam: it allows you to select the specific RAW stream used by the active lens. Unlike most camera apps, which request a simple video stream (e.g., 3840x2160 30fps from a YUV/Video stream), MotionCam asks for whatever RAW settings the camera can provide. This is not the final video crop, but the actual sensor output mode.

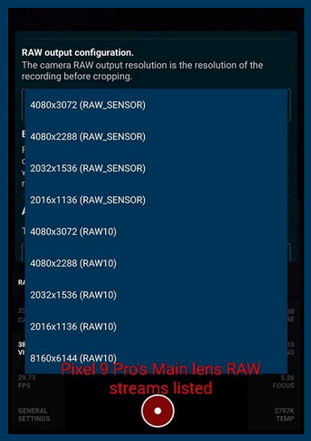
The options available will differ drastically by device. If a device exposes a 50MP RAW stream, you can then crop that for an 8K mode; if it exposes 12-bit RAWs (often designated as RAW12), this is where you enable it. Besides the resolution, you will also see a designation of the stream's container in parentheses.

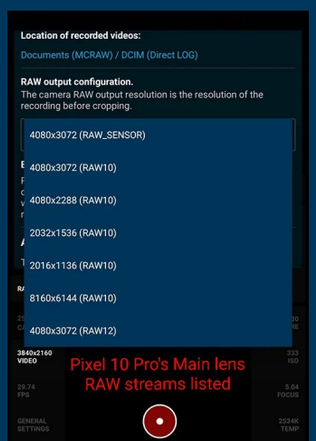
Here is a breakdown of the various stream options you may encounter…

WIDTHxHEIGHT (CONTAINER)

4000x3000 (RAW10)





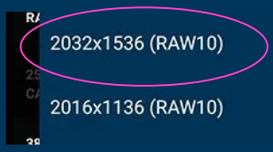


STREAM RESOLUTION – WIDTHxHEIGHT

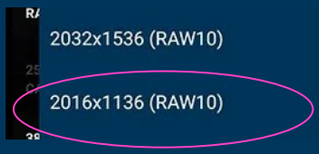
OPEN GATE STREAMS: Open Gate implies the full sensor surface area is being used—you are not cropping anything and are getting the maximum area to shoot with. These are the most common streams for MotionCam, often appearing as the only choice. They typically come in 12MP resolution (around 4000x3000, with minor pixel variances).

OPEN GATE CROP STREAMS: These streams generally offer around 8.9MP, sufficient for 4K video, but achieve this by performing a full sensor area crop, often to a 16:9 aspect ratio (e.g., cropping a 4000x3000 area to roughly 4000x2200). The key benefit of these cropped streams is that they often unlock higher framerate access. For example, if an Open Gate stream maxes out at 30fps, a cropped stream might allow for 60fps or higher.

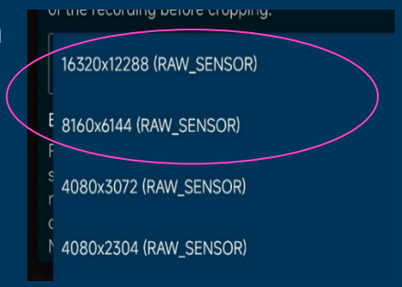
OPEN GATE BINNED STREAMS: These are similar to standard Open Gate streams, but they utilize native binning offered by the device itself to reduce resolution. This conserves both performance and storage. While the app can perform binning for you, some devices offer this natively directly from the stream.



OPEN GATE BINNED CROP STREAMS: These streams are a rare combination of binned and cropped modes, often seen on Google Pixel devices. They are powerful because they tend to allow for ultra-high framerate access (120-240fps), though this comes at the expense of both resolution and a cropped stream area.



UNBINNED OPEN GATE STREAMS: These are rare stream types, also known as the full resolution modes. These are the 48/50MP/200MP RAW modes possible with Quad/Nona Bayer sensors. Use these streams to access the highest possible resolution modes (like 8K), but beware: their performance demand is ultra-high as the app gets +48MP worth of RAW data.



RAW CONTAINERS (RAW##)

This option designates the bit container used for the RAW stream. Understanding the available containers is vital, as using an appropriately sized container will reduce storage and the performance toll on your device.

RAW_SENSOR (RAW16)	16-bit container	The native default, sometimes your only option. **Note:** This does not mean the device shoots 16-bit; it's a large container padded with unused data if your sensor shoots in lower color depth (eg. 10-bit)
RAW10	10-bit container	Preferred choice over RAW_SENSOR if available should your device shoots in 10-bit only, as it significantly reduces storage and performance toll on the device during capture
RAW12	12-bit container	Rare. Appears if your device natively supports 12-bit ADC or DCG modes.
RAW14	14-bit container	A **unicorn**. Extremely rare and generally indicates access to DCG16. Never expect to see this.

Sizing Your ‘Cup’

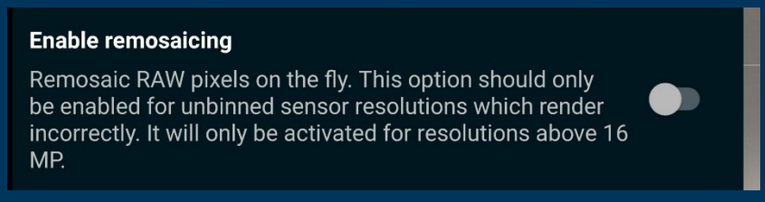
A common question is the difference between RAW10 and RAW_SENSOR. Think of it like this:

Pretend you have a 0.5 Liter Cup (RAW10) and a 0.5 Liter serving of soda (10-bit data)—they are sized perfectly. A 2 Liter Cup (RAW_SENSOR/RAW16) would be great if you had 2 Liters of soda, but if you only have 0.5 Liters to fill it up with, you're now carrying an unnecessarily large cup.

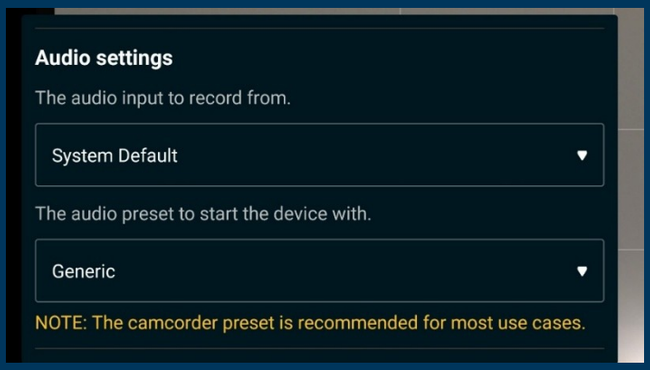
In short, size accordingly. While MotionCam will compress the file, using a container larger than necessary will still tax performance more as the app has to handle more incoming data from the container being padded. If your device shoots 12-bit or 14-bit and only offers RAW_SENSOR, you must use it to harness the higher color depths; otherwise, stick to the smallest container that fits your data. An analogy of RAW10 vs RAW_SENSOR with a 10-bit ADC sensor would be this…

ABCDEFGHIJ000000 vs *ABCDEFGHIJ*

The above example is an easy way to understand the downside of RAW_SENSOR, more space for the same amount of data. For higher color depth, RAW12/14 container options are ideal but not always available.

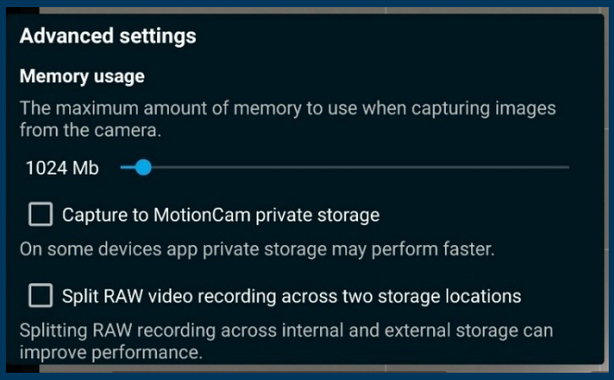


Enable Remosaicing: This option should only be used if after selecting an Unbinned stream, the image appear purple/magenta and the colors look broken. Applies a custom app ‘remosaicing’ algorithm that attempts to re-organize the pixel color array into a correct RGGB grid. Only activates once toggled and when a resolution above 16MP is detected.



Audio Settings: This area allows you to configure both the audio/microphone input source as well as their preset (the audio processing and filtering, such as background noise reduction applied by the device)

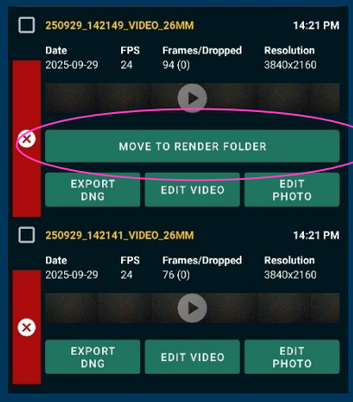
Use the top option to view any external mics connected. NOTE: Bluetooth wireless mics do not work, only wired mics are functional as of now.



Advanced Settings (RAW Capture): This area should be left alone unless you are attempting to extract more performance from the device, whether due to memory or IO bottlenecks you've encountered, or pushing extreme capture settings. It will only appear in RAW video mode.

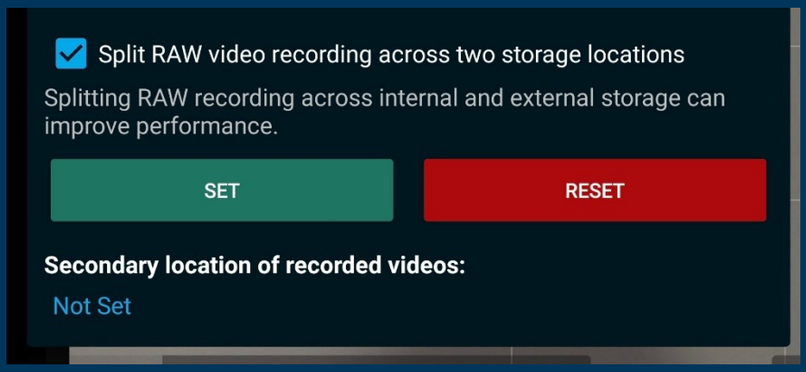
Memory Usage: Allows you to increase/decrease via a slider the RAM allocation that MotionCam will use for RAM buffer that is used to store incoming RAW data. A higher value could increase performance during captures, however it may also impact system performance if you request too much. An overly high value may result in app crashes or noticeably lower capture performance (may also cause longer app boot up times), use with caution.

Capture to MotionCam Private Storage: This setting, which applies only to MCRAW captures, determines whether files are written to your designated folder or the app's 'private' folder (inaccessible at the root/system level).



This option exists to eliminate potential I/O bottlenecks on devices that restrict file write speeds to the standard user-designated folders.

IMPORTANT: Checking this box On (may eliminate bottlenecks) should only be used when absolutely necessary. If files are written to Private Storage, you must manually transfer them out using the MOVE TO RENDER FOLDER option (circled in purple) in the MCRAW Manager. Failing to do so means the files will not appear in a standard file explorer, and deleting the app will wipe your recordings.

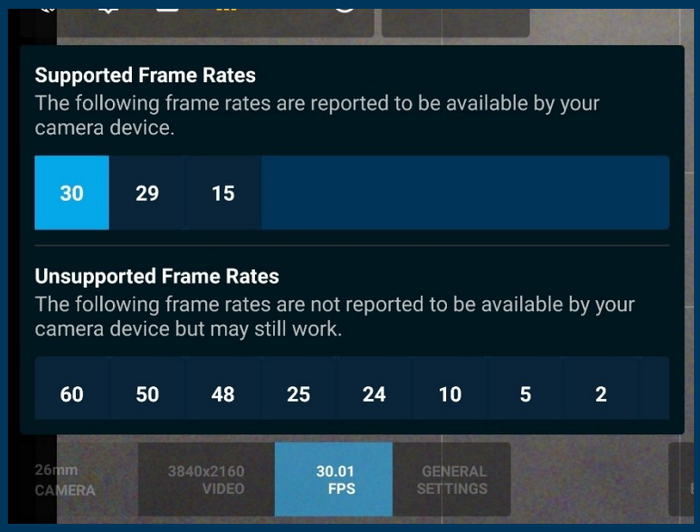


Split RAW Video Recording Across Two Storage Locations: This option allows you to use two different storage locations (e.g., internal and external) to alleviate write speed issues that may occur with slower internal storage modules.

If checked, an additional section will appear below to set the secondary external location and show the designated path. Uncheck the box to disable the feature, or press RESET to erase the path. Note that a designated path is required for the option to be active, even if the box is checked. While this feature was a lifesaver with older storage (pre-UFS 3.1), it's since become less necessary, except for lower end models.

⚠ WARNING: Seriously, Don't Do This!

Haha! So you've decided to designate an internal location as the secondary one as well? Nice try! It won't be faster and will actually tank performance as you've now made the app write to multiple separate internal locations simultaneously. This will impact performance heavily, so please don't do it... Don't ask how I know this...



Framerate: This area on the Camera & Display Controls bar shows the current framerate being outputted in real-time. It displays with hundredth decimal precision (e.g., 000.00fps).

The value may fluctuate if the device cannot keep a steady constant, known as a Variable Framerate (VFR), but a Constant Framerate (CFR) will remain perfectly still. The app will request an exact number but some devices may give slightly variant ones. As an example, if you select 30fps, the lens may instead give 30.01, or 30.10, or even 29.96. These are arbitrary examples and the values may vary strictly per each lens and per device, however it's important to account for them in post to avoid audio drifting.

Crucially, if the selected shutter speed falls below your target framerate, the output will change to reflect the current lower rate. For example, selecting a 30fps slot that gives 30.10 with a 1/30s will turn it red due to the slightly lower shutter vs framerate. Additionally, an example shutter speed of 1/16s will cap your FPS counter at ±16.00fps; the shutter speed value will turn red to warn you of these potential framerate overrides.

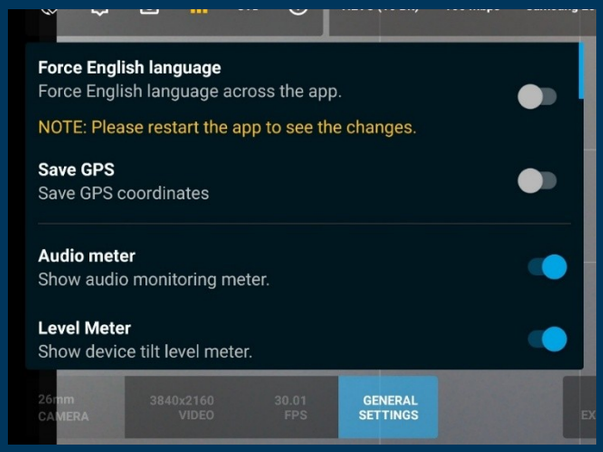
To open the menu and view the available capture options, tap the framerate number on the bottom-left settings bar. The listed options shown will vary by device and are categorized as follows:

Supported Frame Rates: These are the framerates reported by the device's Camera2API as 'officially' supported. They should work unless the OEM has introduced an issue.

Unsupported Frame Rates: These are framerates that the device did not explicitly list as available. This omission does not mean they won't work - it simply means the device didn't mention them rather than outright state they won't run. While compatibility is not guaranteed (and some may cause crashes), you are free to try them. Note: You may need a cropped RAW stream to access higher framerates (see page ## for details).

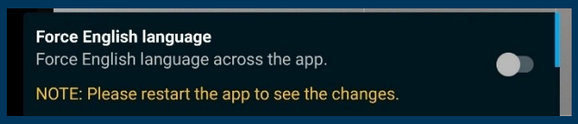
Pro Tip: *Hidden System Priority*

If you try an unsupported framerate that doesn't fully work yet it doesn't crash (e.g., you select 60fps slot but it only runs at 30fps), it's still a win! You've gained a hidden advantage: the higher framerate slot forces the device to prioritize capture resources. This can squeeze more performance from the system and increase the performance ceiling of your capture.



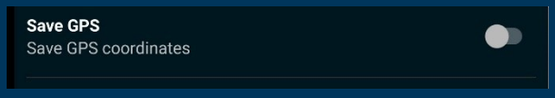
General Settings: Tap on this setting in the Camera & Display Controls bar to open up the General Settings menu.

This menu provides options in relation to overall user interface and viewing guides, as well as additional overlay and tool controls.

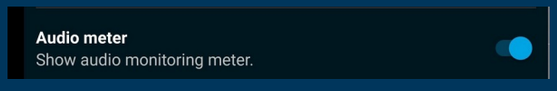


Force English Language: This setting can be used to force English as the language selection when running MotionCam. If disabled, the app will attempt to use the current system language instead if already added under the supported languages list. If you change this setting, ensure you restart the app in order to apply it.

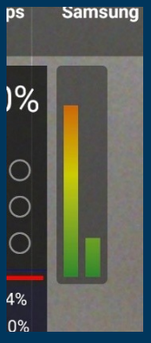
Here's the current list of supported languages: SA Arabic (ar) • DE German (de) • ES Spanish (es) • FR French (fr) • IT Italian (it) • JP Japanese (ja) • KR Korean (ko) • PT Portuguese (pt) • RO Romanian (ro) • UA Ukrainian (uk) • CN Chinese (zh)

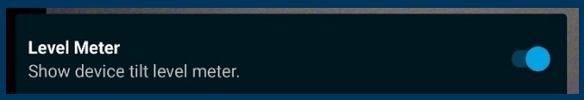


Save GPS: Enable this setting in order to embed GPS coordinates/Location data onto captured MCRAW and DNG files' EXIF information. Please note this option only works if the device location services are already enabled at the time of capture (the app itself can't turn on location services if not on)



Audio Meter: Enable this setting to display a real-time Audio Meter on the upper-left side of the user interface, positioned just to the right of the Histogram if that is also enabled. The meter uses separate bars to indicate the signal strength for both the left and right channels in a stereo arrangement. A lack of bars means no audio is being received, while the presence of a short, green bar indicates a faint audio noise level. As the audio signal increases, the bar's gradient will change towards red to signify a louder noise. If the bar maximizes out, the signal has blown out or clipped, meaning you have exceeded the dynamic range of the microphone currently in use.

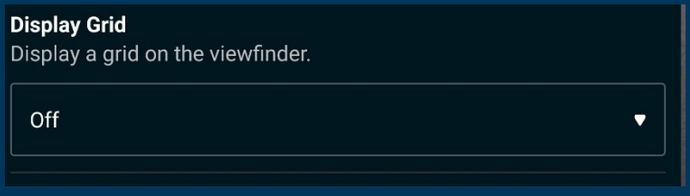




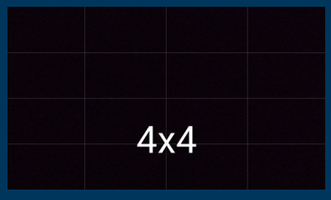
Level Meter: This setting enables an orientation level meter that will appear on the center of the viewfinder.

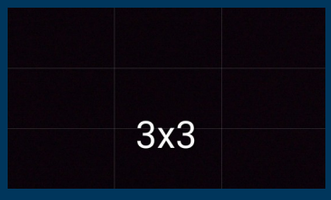
Use this level meter to determine the tilt of your device in degrees. Positive values indicate left tilt and negative values indicate a right tilt with 0.00° being perfectly level relative to the horizon. The Orientation scale used is to the right.

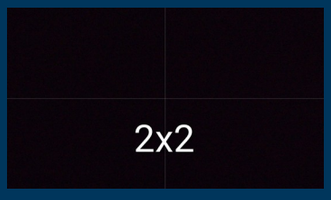
The meter has a central line that tilts with the device showing the degree orientation, and changes with color starting from green to red as the tilt increases more. A left and right squares will perfectly align whenever leveled and the color will become dimmer to indicate a good level orientation. See the blow examples for how it will look.



Display Grid: This setting allows you to enable a display Grid overlay on the UI/Viewfinder. Tap on the rectangle area to open the available selection (Set Off to Disable). Here's the current options…

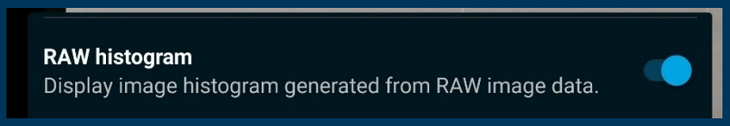






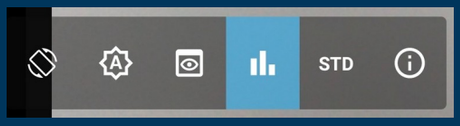
Lock Exposure/Focus/White Balance on Record: Toggling these options on will make the app automatically lock the current settings of their respective categories once you begin recording,

You can undo the manual locks triggered as per usual once recording, however take note that these toggles will persist even after app is closed.



RAW Histogram: This toggle will turn On/Off the RAW Histogram on the top left of the viewfinder.

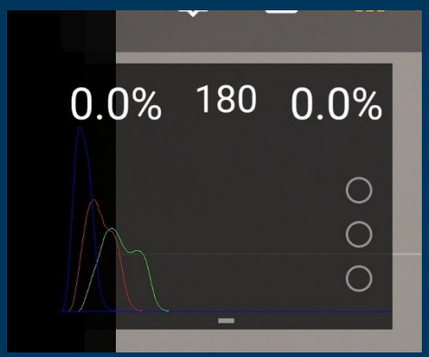
(Note: This option can also be toggled on the top left Histogram Quick Toggle button)



The histogram is one of the most powerful tools provided by MotionCam. It is strongly encouraged to master it in order to fully extract the app's imaging potential.

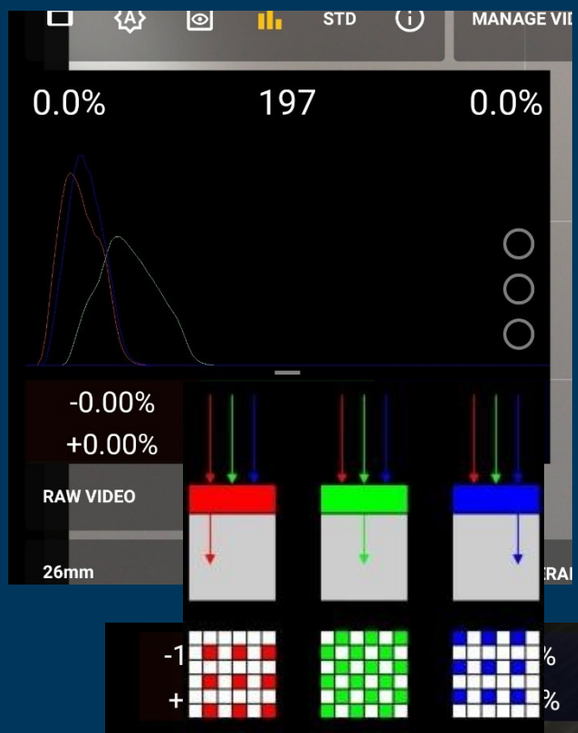
Practically all other app histograms provide you a JPEG histogram, meaning that you get the information data that would occur after saving to a JPEG. This is problematic as JPEGs are lossy containers operating on an 8-bit color depth, hence not able to properly contain the deep highlight and shadow information that would present itself in a full +10-bit RAW image as most modern sensors will produce. What this translates to is having to guesstimate what the limits of your dynamic range are, without accurate precision (eg. JPEG histogram may show clipping, but the RAW data may still have plenty of space left to keep pushing further).

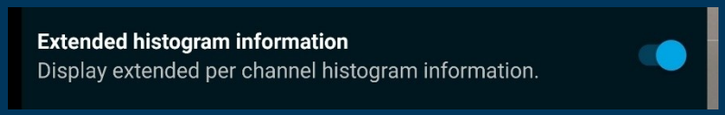
With MotionCam, you obtain the literal sensor data as would be received in a RAW image container. What this means for you is that the accuracy levels provided by the MotionCam histogram are second to none. Due to the many complex aspects of the histogram and the data it provides to the user, a full comprehensive section on its usage can be found separately on page ## (jump to page ## for a quick breakdown if you just need the essential info).



You can tap on the histogram to make it larger (and tap again on it to shrink it).

This mode will display the histogram without a see through background to ensure charts are visualized accurately. It will also force on the Extended Histogram Information regardless if the setting is Off (see next page). This override will not carry back once you shrink the histogram back.



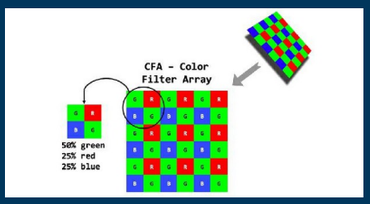


Extended Histogram

Information: This toggle enables an additional section of the histogram to display detailed color channel information. To understand your sensor, it's essential to know that it is built on a Bayer pattern where each pixel receives a Red, Green, or Blue filter. Enabling this option adds a bottom area that shows the individual Red, Green, and Blue channel data, allowing you to view the full range of the sensor's color space.

This feature uses percentages to indicate when separate channels are exceeding their own dynamic range:

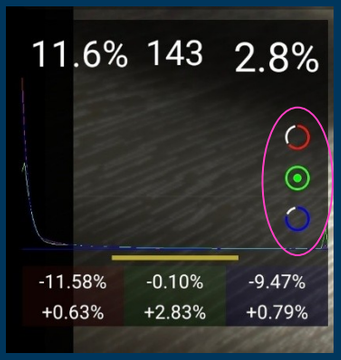
Positive values (+) indicate the percentage of the sensor channel that has clipped (highlights that exceed the channel's range and are now pure R/G/B).

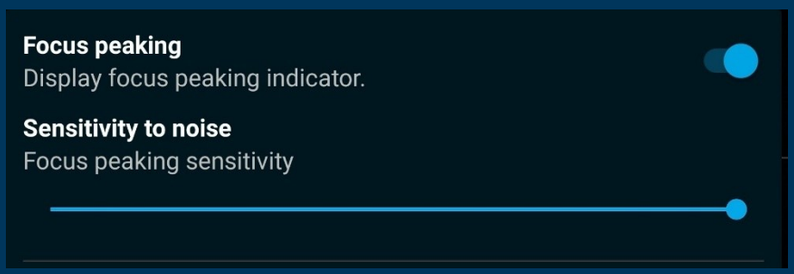


Negative values (-) indicate the percentage of the sensor channel that has crushed (their area is reading pure black and holds no more data).

Because different color channels have varying light sensitivities (for instance, green pixels take up 50% of the sensor and often receive more light), they will frequently crush and clip at different thresholds. This powerful feature allows you to carefully assess the scene with absolute precision and can even be harnessed for highlight recovery/reconstruction by intentionally clipping only individual channels. For a comprehensive explanation of histogram operation and how to best harness this function, jump to page ## (and see page ## for more on highlight recovery).

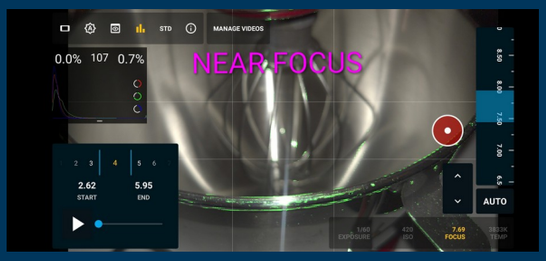
NOTE: The 'traffic lights' that appear to the histogram are a way to quickly review the color channel highlight clipping for an area of up to 1.00% (+0.00% will be light off, +0.01% to +0.99% will begin circling light), anything equal to or beyond +1.00% will be indicated by a fully circled and illuminated light of its respective color channel

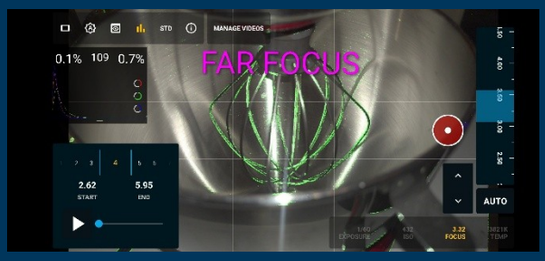




Focus Peaking: This setting enables a focus peaking overlay that displays green lines around surfaces in the scene that are currently in the sharpest focus. This is a

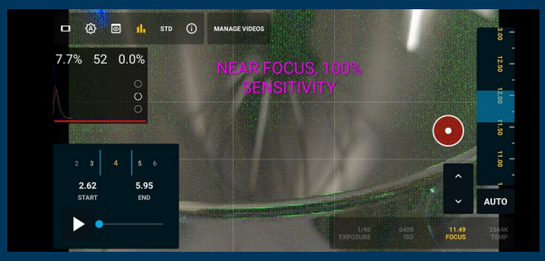
critical tool for fine-tuning focus and can be used with either manual or autofocus modes. The overlay will only appear once you have opened the focus controls by tapping the Focus value on the bottom-right bar and opened its corresponding slider rack.

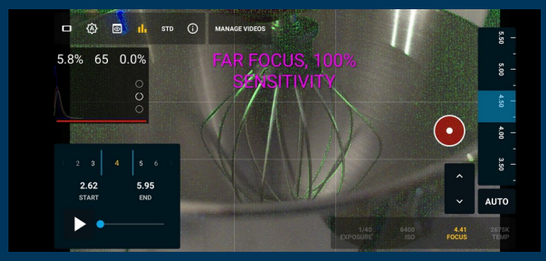


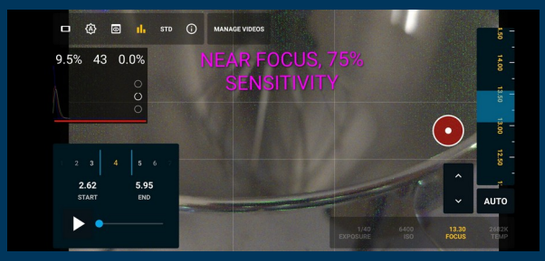


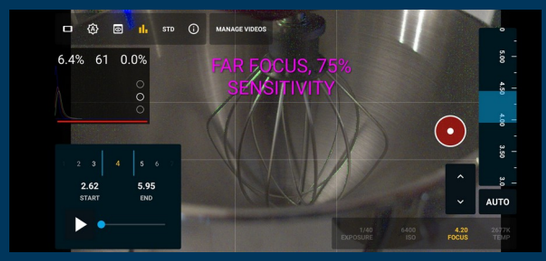
Sensitivity to Noise: This setting works in conjunction with Focus Peaking to adjust the sensitivity of the green overlay. Depending on the Viewfinder Resolution (see page ##) and the environmental lighting (especially in low light) sensor noise may induce false positive data by showing peaking patterns on surfaces that aren't actually in sharp focus. You can reduce the sensitivity to mitigate this noise, but at the cost of a less prominent overlay.

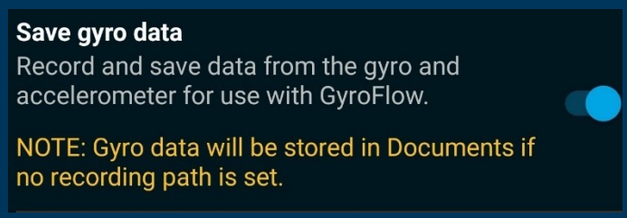
Tweaking is encouraged to find a suitable balance, as too much sensitivity will introduce significant artifacting on the viewfinder, while too little will make the overlay almost non-existent.





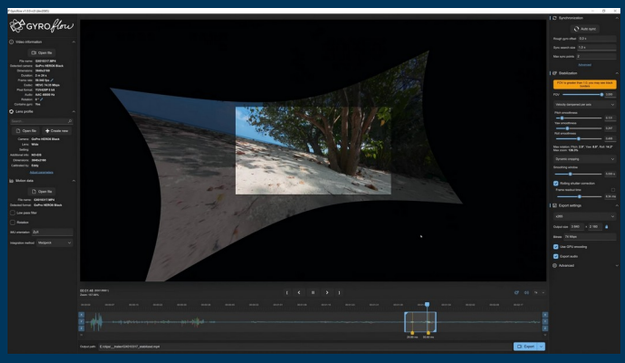






Save Gyro Data: Toggle this setting On to have the app record and save gyroscopic and accelerometer data (the movement sensors on your phone’s readouts) into a .gcv file required to use Gyroflow stabilization.

NOTE: YOU MUST TURN OFF OPTICAL IMAGE STABILIZATION IF YOU INTEND TO USE GYROFLOW SINCE THE MOVING LENS ELEMENT IS NOT ACCOUNTED FOR IN THE SENSOR DATA!!!



Gyroflow is a program that can apply superior camera digital/electronic image stabilization than the one otherwise present on phones or full sized cameras.

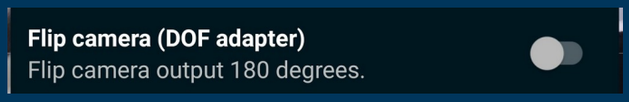
It works via calibration to your specific camera lens, which allows it to apply things such as distortion correction and de-warping, as well as it can fix things like rolling shutter. More importantly, this is then combined with its movement based stabilization algorithm that leverages your accelerometer /gyroscopic sensor data to measure the specific scene displacement rather than just visually guessing the movement that occurred.

The resulting shake reduction is nothing short of spectacular, however do note it's generally more effective with faster shutter speeds (180° rule lovers beware) as well as it will work best with wider lenses. As with all forms of digital stabilization, you can expect some cropping too, however this is adjustable.

If you haven't already, I would also encourage to check them out and try it for yourself as its an incredibly effective tool! They also have an amazing community and their developer is super responsive – they're like long lost cousins to the MotionCam community.

https://gyroflow.xyz/

https://discord.gg/6zyUExsXe



Flip Camera (DoF Adapter): Use this toggle to flip the viewfinder upside down/by 180 degrees.

This setting is extremely useful for whenever you use an external lens or Depth of Field adapter which will generally provide an upside down image projection; this is also the case for some

lenses that have less prism elements and do not correct the light intake. This results in an inverted light input which can make viewing difficult in both cases.

This option does not impact the footage captured and is only applied at the viewfinder level. Also do note that at the moment it does not work with Direct Preview, only Android Preview.

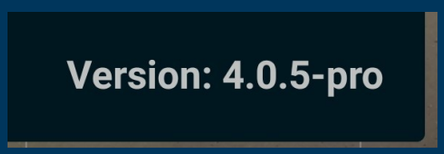
Squeeze Viewfinder: This section allows to further adjust the viewfinder squeeze. This is particularly useful for whenever you use external lenses such as anamorphic adapters, which will squeeze the image due to the inherent nature of their distortion. This will naturally warp the preview, use these setting therefore to 'unwarp' the preview.

Any changes done on this area will not impact the capture data and are only meant to assist in the preview. Also, worth mentioning, it will not work with Direct Preview either.

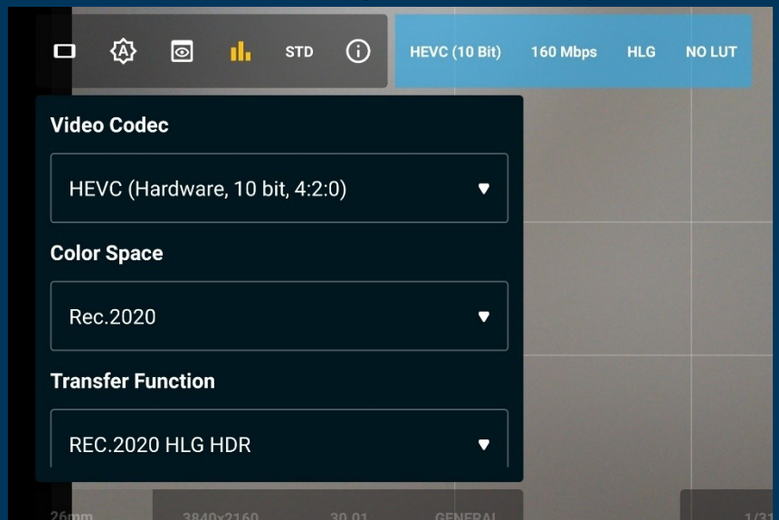
The top slider controls the Vertical Squeeze and the bottom slider controls the horizontal squeeze. Slide left to desqueeze and right to squeeze on both. The lowest/leftmost value is 1.00 representing no squeeze whereas the highest/rightmost value is 2.00, representing a full squeeze by a factor of x2

The bottom numbers provide popular anamorphic lens squeeze ratios such as 1.33, the V represents Vertical and H represents Horizontal to advise you of which aspect is being squeezed by the preset.

Press on the RESET button to undo any slider or preset changes applied.



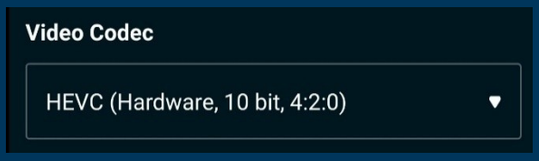
App Version (Easter Egg - Command Prompt): This area allows you to verify which app version you are on specifically. Still reading? Well, press and hold it to see something cool, the Command Prompt! This feature is still a work in progress. It's hoped one day you'll be able to use it for even more advanced system level app functions! For now, not much to report however... 😁



DIRECT LOG VIDEO SETTINGS MENU

Please note that the next following UI Elements are only relevant to the Direct Log Video mode and this settings bar will NOT be applicable nor relevant to the other modes.

To open the Direct Log Settings menu, tap on the top bar displaying the encoding settings (any of the 4 categories will open the same menu), the scroll up/down to navigate the available options.

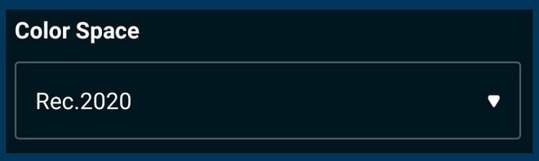


Video Codec: This setting displays the current encoder selection, showing its Codec Name, Acceleration Method, Color Depth, and Chroma Subsampling in a clear format: CODEC NAME (ACCELERATION METHOD, COLOR DEPTH, CHROMA SUBSAMPLING). Press the rectangle box to open the full list of options available on your device.

Hardware accelerated codecs (such as APV, AV1, and VP9) will only appear if a specific encoder is made available by your device. However, software accelerated options like ProRes will be available regardless of the device. Be warned that the ability to reliably encode and handle these is entirely dependent on your device's processor and optimizations.

NOTE: The HEVC encoding used here is not the same as the one used by standard video apps, so your performance mileage will vary significantly. Standard video apps rely on the device's highly efficient Image Signal Processor (ISP) and video stream, which are optimized for speed at the cost of image quality. MotionCam, however, uses its own brute force pipeline to capture and process maximum fidelity. Do not use the performance of standard video apps to measure your expected performance in MotionCam, as this quality-first approach is vastly more demanding.

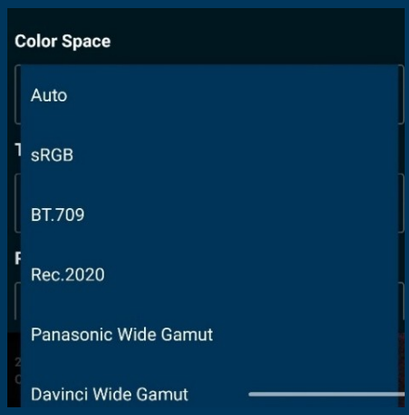
Additionally, if you do not see the 10-bit option available for HEVC, it may be due to ROM or OEM limitations imposed on your device.



Color Space This setting allows you to select the color space to be used in the encoding container. For those unfamiliar, the color space defines the abstract range of colors that will be captured—not necessarily how the colors are shown. This selection is directly impacted by the color depth in use.

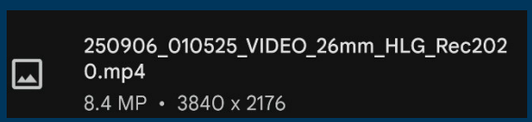
On the accompanying chart, you can observe how the selected color space (represented by the triangles) compares to the full range of human vision (the colored area). For example, a massive space like DaVinci Wide Gamut extends even beyond human perception in some areas. Do note that no sensor can capture the full vision range of the human eye. In fact, many camera sensors capture colors unperceivable by humans because the wavelength filtering characteristics of their color filter arrays (CFA) can never truly match the human eye's response, not to mention issues like different crosstalk behavior (crosstalk being the spillover of light from one color channel's sensor well into an adjacent color channel). However, these outlier colors are often only achieved with artificial LED colored light.

When choosing, you must select an appropriate color space relative to your needs and your Transfer Function (see next section). A larger color space is not always better, as it requires more data to store and will run into complications if the viewing display cannot properly reproduce it. Therefore, wider gamuts above Rec.2020 primarily benefit log video captures, allowing you to retain more color data in certain intense conditions with super deep colors that may challenge the sensor, or helping you match your log captures' native color space requirements (e.g., Panasonic Wide Gamut + Panasonic VLOG).



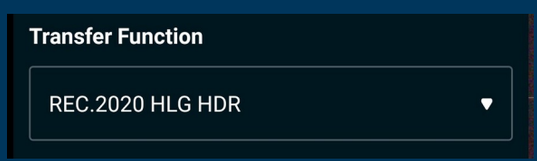
Crucially, to obtain a final delivery-ready video, only BT.709 or Rec.2020 HLG/PQ are recognized options. If you encode using a camera-specific color space like S-Log3's native gamut, it will not be labeled as such in the video metadata, and standard video players will misinterpret the image.

The Auto settings will default to the most appropriate color space (Generally BT.709 or BT.2020). The selected color space will also be stated on the output file's name extension to ensure you do not forget the selection when working with the file..

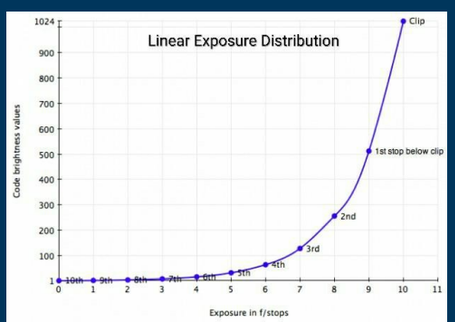


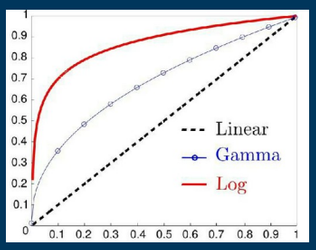
The current list of Color Spaces available in the app is as follows...

sRGB – BT.709 – BT.2020 – Panasonic Wide Gamut – DaVinci Wide Gamut – ACES AP0 Canon Cinema – Sony S-Gamut3 (Cinema)

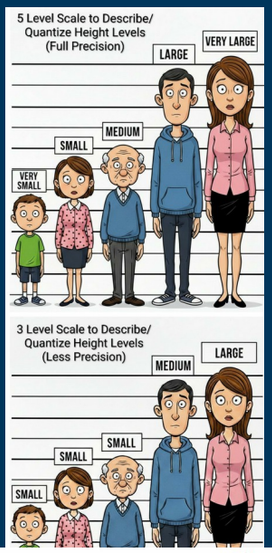


Transfer Function (Gamma/Log): This setting controls the Transfer Function applied during encoding, determining how light information is recorded. It's essential for Direct Log as it maps the scene's dynamic range to your final file. If this sounds like gibberish, stick with me—it's key to unlocking this option!





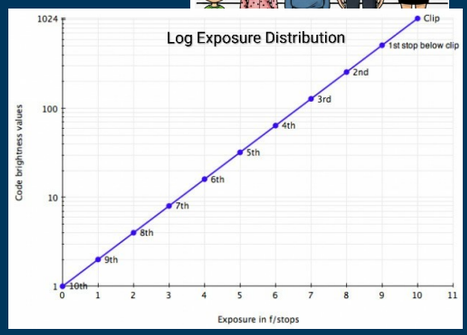
A linear function (the starting point for RAW data) records light precisely. Why is this system terrible for encoding? In 10-bit (1024 total values), roughly 512 values (half of what's available!) are used for just the single brightest stop of light before clipping. The next stop down uses half of the remaining precision, and so on. This drastically under-prioritizes shadows, leaving barely any numbers for dark detail. 'But Ragu! What the hell does this jargon mean!' I hear you asking. Check the top right illustration to see the 'linear' light representation.



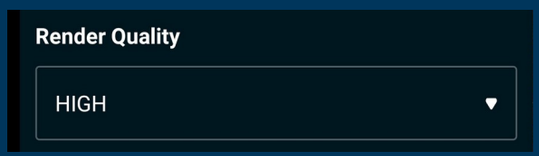
Because of this extreme allocation, Linear data is prone to rounding errors and precision loss in the shadows after lossy encoding, which is why transfer curves are applied: to protect those details from being crushed into pure black during compression.

The familiar (Gamma) Rec.709 curve is designed for SDR displays and does not clip highlights by design, that's a myth! The real issue is that Rec.709 is often paired with 8-bit precision. Since 8-bit uses only 256 possible values (vs. 1024 in 10-bit), rounding errors become much more visible. Rec.709 is fine, but its combination with limited 8-bit precision makes it unsuitable for heavy grading, often leading to color banding.

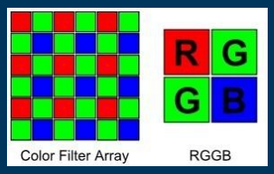
This is where Log functions come in. They are logarithmic—yes, that math! The Log curve mathematically re-distributes the linear values for efficient storage. It intelligently increases precision for shadow information (protecting data) while slightly decreasing it for highlights. Look at the right graph now: Log's dynamic range warping provides a much more equal and discernible representation in the container, giving us better shadow detail (worth the slight highlight compromise!). In other words, the exposure distribution of linear is logarithmically bad, and the exposure distribution of log is linearly good.



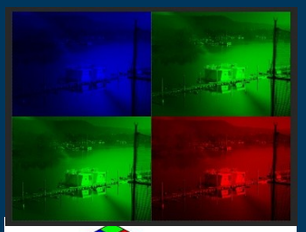
For editing, Log is your best choice for maximum data retention (e.g., S-Log3 + ProRes LT/Standard or 10-bit HEVC + HLG). Remember: if the math is right (big IF) the transfer curve's purpose is only efficient storage. Converting linear data to Rec.709 results in the exact same image as converting linear data to Log, and then to Rec.709. This only concerns transfer functions, though; color space transforms are a whole other complicated topic for later!

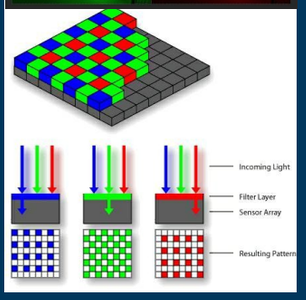


Render Quality: This setting controls the quality of the demosaicing algorithm used exclusively in Direct Log Video mode. You have two settings: Standard, which applies a lower quality algorithm for better performance, and High, which uses a superior algorithm for quality at the cost of processing speed. This choice is irreversibly hard-baked into the final encoded output, regardless of the codec or transfer function, and will not affect the quality of RAW or MCRAW files.

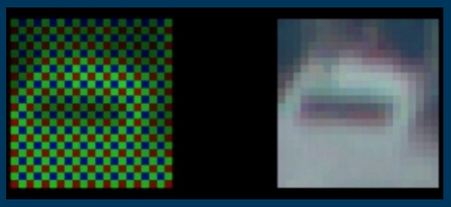


To grasp this, you have to think about your camera's sensor. It uses a Color Filter Array (CFA) in a pattern of Red, Green and Blue within a cluster of 4 pixels (like RGGB, BGGR, etc.) where neighboring pixels never share the same color. This is a clever trade-off to capture a color image with a single sensor, also giving the Green channel a higher distribution because the human eye is more sensitive to green light. The catch is that every pixel is only capturing one color, creating three incomplete images (see the right illustrations)



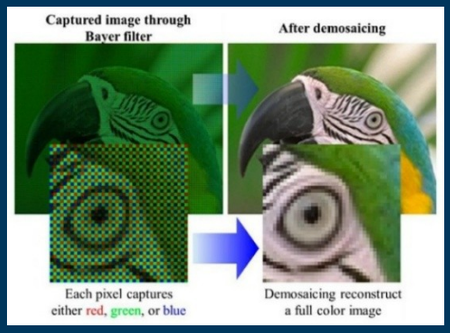


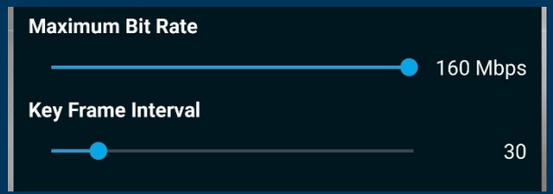
This is basically like when you put on those old 3D glasses! Your amazing brain combines the incomplete red and Blue images from each eye and renders the full color regardless.



Demosaicing is the sensor version of that process. It's the algorithm that interpolates (or "guesstimates") the full color of a pixel by using the information from its single-colored neighbors. The better the algorithm you choose, the better the final color reproduction!

All RAW files contain the full Bayer data, which is why editors can use newer, better demosaicing algorithms later. Since you're encoding directly here, your demosaicing choice is critical as it's irreversible on encoded output files! While this hard baking won't impact some codecs as badly, it remains a quality factor to consider!





Maximum Bitrate: This slider defines the maximum Bitrate allowed during capture for codecs that offer granular selection. Bitrate is the data density per second after compression is performed, measured in Mbps (Megabits per second). Slide right to increase and left to decrease it. The chosen value is shown to the right.

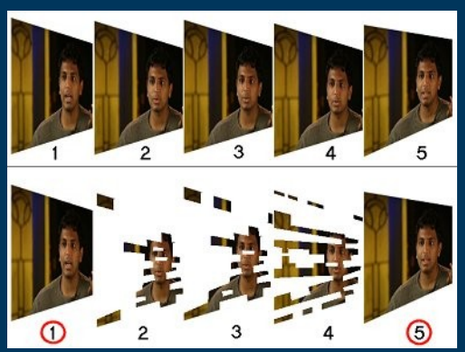
Please note that bitrate allocation is generally not constant, so this option designates the maximum value; the captured output may fluctuate depending on the scene's complexity and the device encoder parameters (firmware level).

A higher bitrate will likely tax performance further but attempts to preserve more data, so you must balance performance against quality. Noisier environments or higher resolutions will inherently require higher bitrates to store the necessary data adequately.

Key Frame Interval: This setting will only appear alongside codecs providing Bitrate selection and allows you to select the interval between Key frames and the intermediate delta/predictive frames. Slide right to increase and left to decrease it. The chosen value will appear to the right. Although devices won't always respect this setting, it's an important tool in compression.

Key frames (or I-frames) are essentially images in the video sequence under which the full data of the image is encoded. The frames that follow (delta frames) only encode the partial data that may have changed since the last Key frame, recycling the rest of the image. This is a crucial effective method to reduce file size.

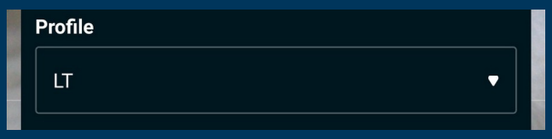
You can observe this efficiency in the image to the right:



The top row uses a Key Frame Interval of 0 (no gaps/intervals). Every image is encoded in full. This is inefficient in scenes with low motion (like a static talking head) and wastes bitrate.

The bottom row has an interval of 3 (3 delta frames between key frames). You can see how only parts of the image that changed between frames are encoded, saving data.

This process significantly saves space, but if the scene is too dynamic (high motion), a high interval can negatively impact quality; there may not be enough data encoded in the delta frames to fully capture all of the new movement. Finding the right balance is key!



Profile (ProRes only): This setting will only appear if you select ProRes under the available codecs. Profile selection replaces both the Bit Rate and Key Frame Interval options that would otherwise appear. Press the box to open the list of available ProRes quality profiles. Key frame selection does not apply in ProRes because it is an intra-frame codec, meaning all frames are encoded in full, independently.

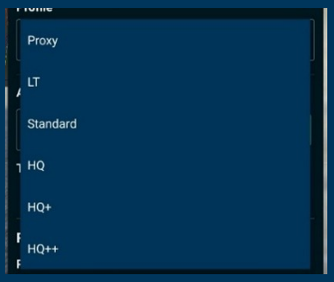
ProRes is offered in multiple profiles with varying levels of data density per frame. This is its key difference: Unlike other codecs which are given a maximum Bit Rate per second and work under that regardless of frame rate or resolution, ProRes profiles designate a specific target

ProRes target data rates

Dimensions	Frame Rate	ProRes 422 LT		ProRes 422		ProRes 422 HQ	
		Mb/s	GB/hr	Mb/s	GB/hr	Mb/s	GB/hr
UHD 3840 x 2160	24p	328	148	471	212	707	318
	25p	342	154	492	221	737	332
	30p	410	185	589	265	884	398
	50p	684	308	983	442	1475	664
	60p	821	369	1178	530	1768	795

Bit Rate that adapts on the amount of frames and resolution received. *NOTE*: The actual Bit Rate captured will remain variable, but it will try to work around the guideline target of the profile, this allows the encoding to adapt more predictably based also on the scene. Additionally, light or noise may drastically impact the size given the lack of ISP noise reduction (eg. iPhone ProRes). Remember, Noise = Data!

A higher quality profile simply targets a higher bit rate that scales proportionally with your given resolution and frame rate. In other words, increasing resolution and/or frame rate will increase the target bit rate to match the added information being recorded. For example: 4K ProRes 422 will provide a target bit rate of around 589 Mb/s at 30fps, compared to 492Mb/s at 25fps. The target will also increase or decrease proportionally with changes in resolution.



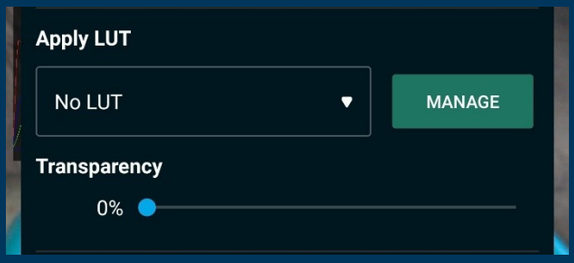
You have multiple ProRes Profiles to choose from…

ProRes Proxy: Originally meant for editing proxy purposes. Not recommended due to its very low target bit rate. While it's easier to edit than HEVC because it's compressed far less densely, it provides much less data relatively speaking. Best avoided, if possible.

ProRes LT (Lite): This profile provides a good balance between file size and strong performance. It is a great starting choice and the recommended baseline profile for ProRes usage in order to obtain quality comparable or superior to 10-bit HEVC.

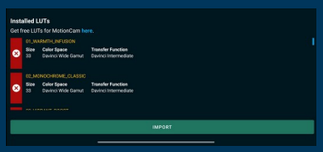
Prores Standard: Although the "Standard" nomenclature simply refers to the fact that this is the normal 'ProRes 422' without compromise, it is the recommended option if you have extra processing power to spare. It provides great data density with less compression than the performance-prioritizing LT profile..

ProRes HQ (HQ+/HQ++): The highest quality of ProRes 422, this version provides the least amount of compression in a visually lossless sense. It is best reserved for the highest visual requirements as it is very resource-intensive. This profile is only recommended if your device has a state-of-the-art, flagship-grade processor and excellent thermal performance. For lower end devices, pixel binning mode can be a great compromise to allow reliable ProRes HQ captures. The HQ variants (HQ+ and HQ++) simply unleash the target bit rate more and more to increase quality potential. **Beware**: if the target becomes higher than your device can handle (raison d'être of + and ++ increments), this will likely result in audio syncing issues for the capture. Testing beforehand is strongly advised to ensure no capture issues with your specific model.



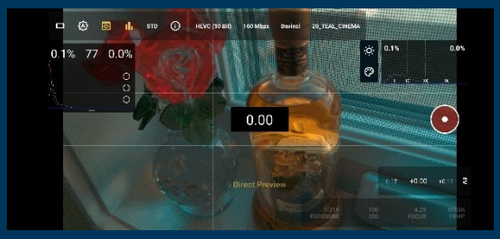
Apply LUT: Use this box to select the LUT (Look Up Table) to be baked into your Direct Log capture footage. Tap on the box to expand the list of LUTs already added. As of this current release, do be mindful that having a LUT in the active selection **hard bakes** the Look into the footage, so it's important to disable it before capture if you intend to use them for preview only.

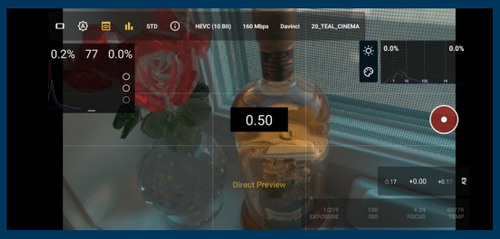
Note: Applying a LUT will immediately override the Transfer Functions and Color Spaces applied before hand and match them to the LUT functions selected upon initial import.

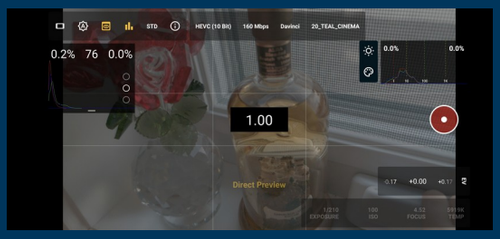


MANAGE: Press on the green MANAGE button to open the LUT manager and scroll through/view/add/remove LUTs (more on this one in page ##).

TRANSPARENCY (Blending): Use this slider to adjust the intensity of the LUT overlayed/baked into the image. You can essentially create a hybrid image without completely integrating the LUT into the capture and adjust it to taste. Slider right to increase and left to decrease. The numerical percentage shown on the left indicates the Transparency amount, with 0% indicating the selected LUT is being applied fully with no transparency, 50% indicating a half-LUT and half-native transfer function image fusion, and 100% showing the LUT is completely transparent and not visible in the image.

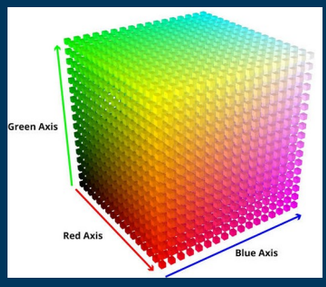


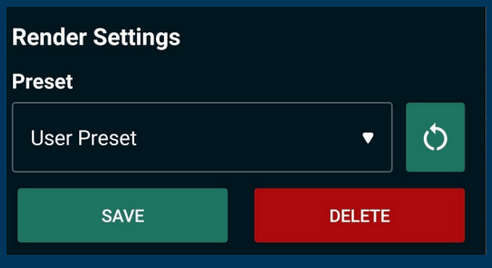




LUT Summary, for the uninitiated

A Look-Up Table (LUT) is a pre-set conversion guide; a mathematical instruction sheet that transforms every pixel's color, brightness, and contrast to manipulate the output into a desired look. If you think of grading like adding sauce and seasoning to food, imagine making your own sauce (you mash up a basket of fresh, raw tomatoes, add vinegar, etc). – a LUT is the equivalent of a ready-made bottle of ketchup instead: it instantly and consistently transforms the raw ingredients into a final, flavorful, and palatable product (like vibrant Rec.709 for example). LUTs are great for monitoring and/or getting an immediate look. That said, the best colorists make their own ketchup from scratch sometimes (custom grading) rather than relying strictly on a 'pre-made bottle' only, as a rigid LUT applies the same transformation regardless of the unique data in your scene. They are excellent tools but should be used wisely.

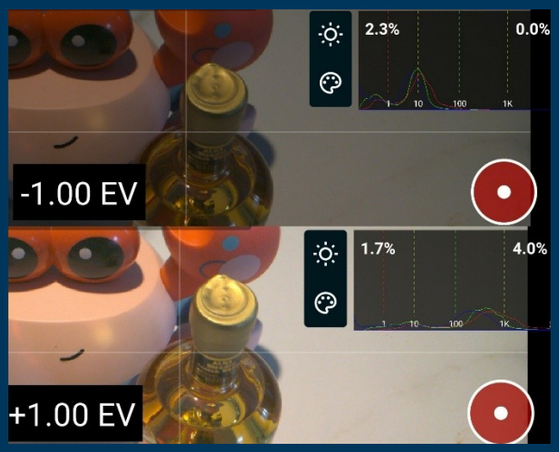


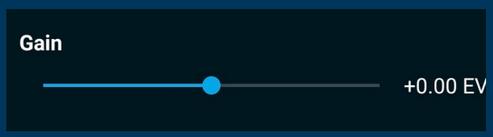


Render Settings: This section allows you to alter the encoding pipeline and how the Direct Log outputs are rendered visually. It provides you with a large selection of image adjustments. Any changes applied can be undone by pressing the circle arrow. To save a set of changes, press the SAVE button; a prompt will appear allowing you to write the name of your preset. To delete this preset, select it on the drop-down box list and press DELETE.

All changes applied in this list can be viewed and adjusted real-time while using Direct Preview mode (on the respective sliders, hold and slide them to transition into real-time adjustment mode).

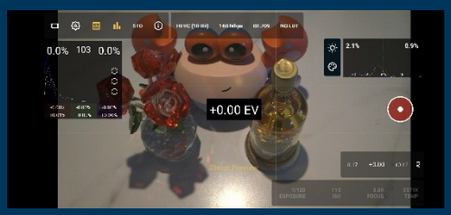
Additionally, use the encoder histogram to identify their impact at the data level for maximum precision; you can see an example in the right image comparing +1 EV vs -1 EV (RAW data and capture are identical, however the encoded data is being treated completely different). We will touch more on this later as this will be important while capturing using Log transfer functions in order to ensure the full container tonality is being leveraged

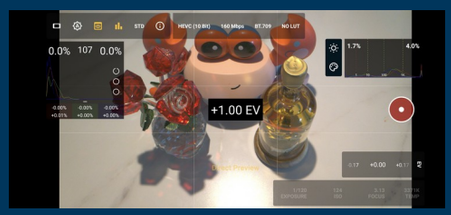


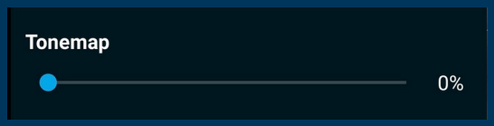


Gain (Digital EV): This slider provides you with control of the encoding digital exposure compensation. It allows you to boost or reduce the exposure of the raw data in order to better accommodate it into codec containers which may have limitations in their dynamic range, of how they store the data.

Of all the sliders, Gain control is perhaps the most important one as it plays a significant role when capturing in Log (more on this later, jump to page ##). A value of +0.00 EV is the default state and does not alter data. Slide right to increase and left to decrease.





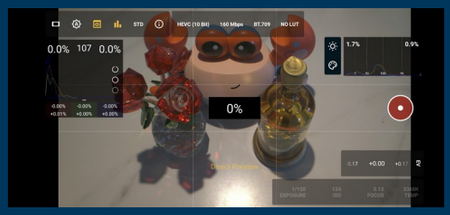


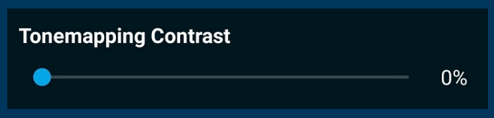
Tonemap: This slider provides you with the ability to control the tonemapping amount applied during encoding. A 0% value is the default state and indicates the setting is not engaged, while a 100% value indicates the maximum tonemapping available is being applied. Granular controls of the intensity can be achieved with values in-between. Slide left to increase and right to increase. This setting will impact Direct Log performance as it requires more resources to run it.

Tonemapping is essentially the ability to compress a wide dynamic range into a smaller one for better viewing (Eg. 10-bit captured data into an 8-bit SDR), it can particularly useful to properly

present a scene that may otherwise look flat if the dynamic range is exceeding the available tonality and can bring back the proper look and feel of a scene

Note: the Tonemap setting must be active (A value of ±1%) for the next two settings (*Tonemapping Contrast/Saturation*) to also work and the intensity applied will directly impact their effect

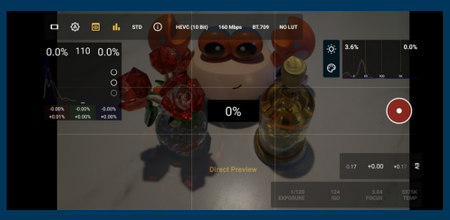


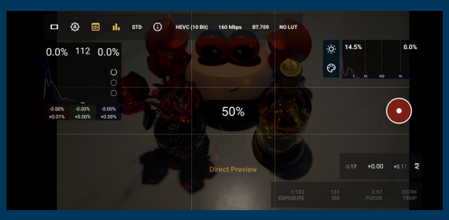


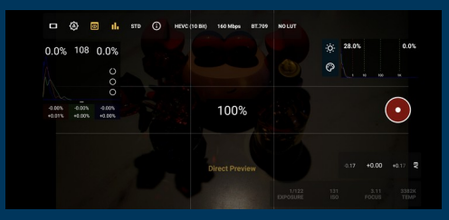
Tonemapping Contrast: This slider allows you to control the contrast intensity levels applied when Tonemapping is enabled.

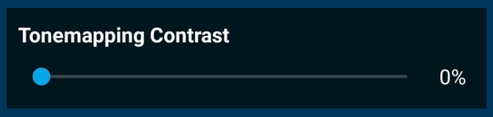
A 0% value is the default state and indicates no changes to the tonemapping contrast levels while 100% indicates the highest possible intensity is applied.

Assuming Tonemapping is On, this option will adjust how the contrast levels are handled on it. The below examples used the Tonemap setting at 100% to show their maximum effect







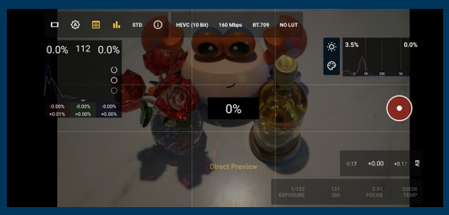


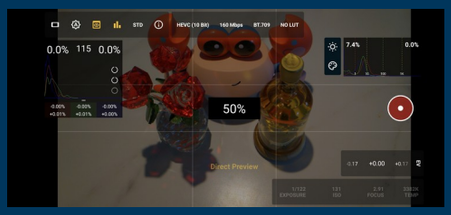
Tonemapping Saturation: This slider allows you to control the saturation levels applied when Tonemapping is enabled.

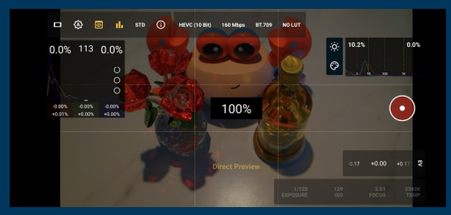
A 0% value is the default state and indicates no changes to the Tonemapping Contrast levels, while 100% indicates the highest possible intensity applied.

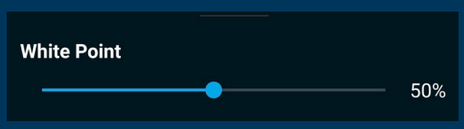
Assuming Tonemapping is On, this option will adjust how the saturation levels are handled on it. The below examples used the Tonemap setting at 100% to show their maximum effect.

NOTE: This Saturation setting is independent of the encoding Saturation itself, and will potentially compound on its effect if both are altered in-tandem. This saturation slider pertains to the tonemapping engine itself.



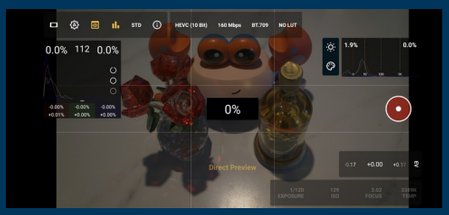


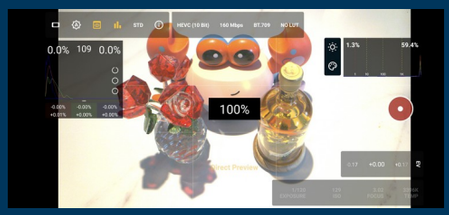


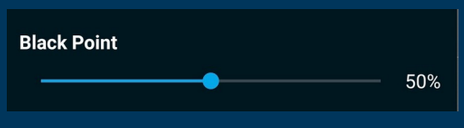


White Point: This slider let's you control where the 'absolute white' point is and will impact how brighter tones are handled.

A 50% value is the default state and indicates no changes to the White point. Slide right to increase the white point up to 100% (maximum offset) and 0% to decrease it (lowest offset).

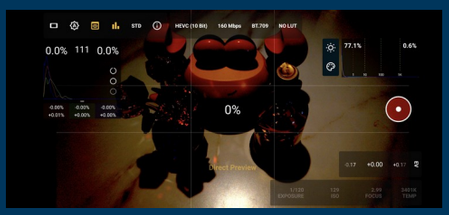


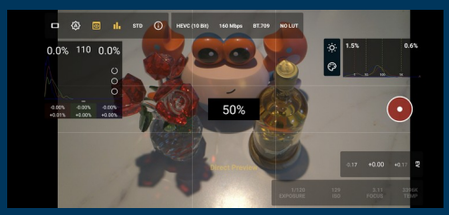


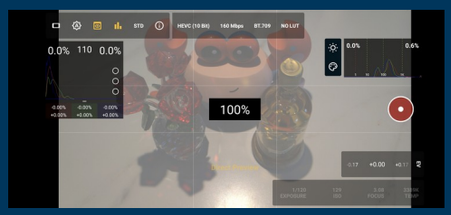


Black Point: This slider let's you control where the 'absolute black' point is and will impact how darker tones are handled.

A 50% value is the default state and indicates no changes to the Black point. Slide right to increase the black point up to 100% (maximum offset) and 0% to decrease it (lowest offset).

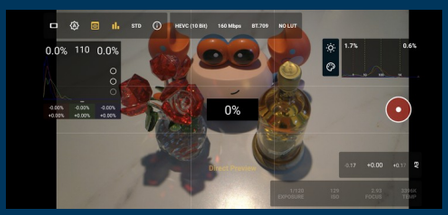


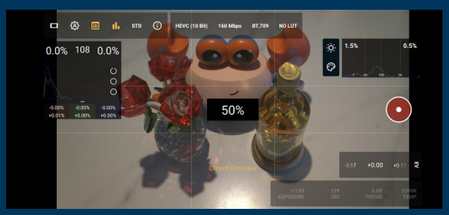


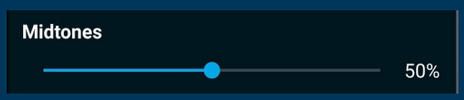


Shadows: This slider let's you control how shadow details are rendered, allowing you to boost or suppress them accordingly

A 50% value is the default state and indicates no changes to the Shadows point. Slide right to boost shadows up to 100% (brightest) and 0% (dimmest) to decrease them.

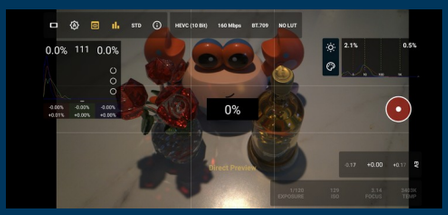


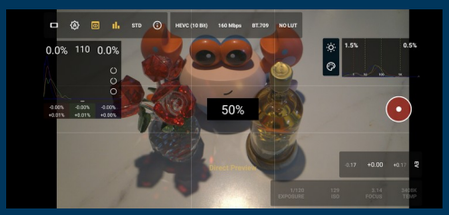


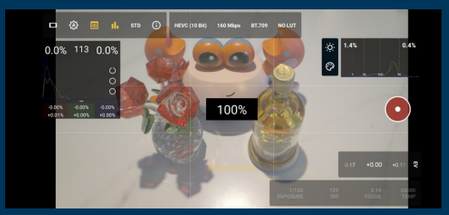


Midtones: This slider allows you to shift the midtones (middle region of data that's neither dark nor bright where the majority of light data lies) and how they're rendered.

A 50% value is the default state and indicates no changes to Midtones. Slide right to increase the Midtones up to 100% (maximum offset) and 0% to decrease them (lowest offset).

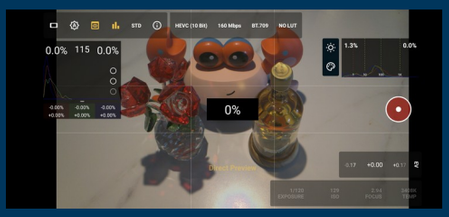


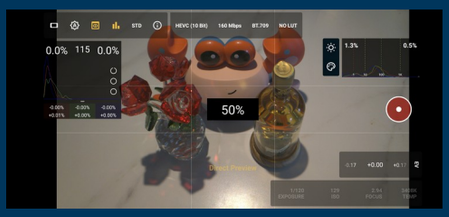




Highlights: This slider allows you to adjust the intensity of the highlights captured and how they're rendered.

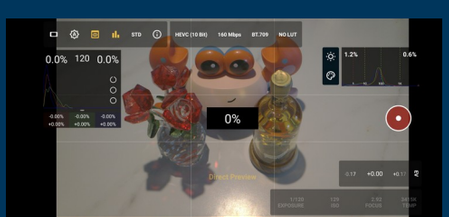
A 50% value is the default state and indicates no changes to the Highlights. Slide right to brighten highlights up to 100% (maximum offset) and 0% to flatten them (lowest offset).

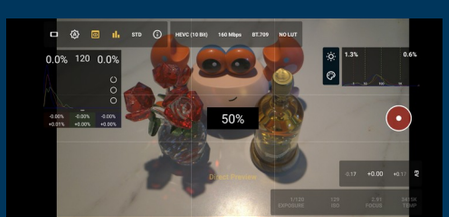


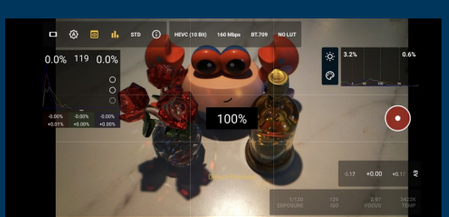


Contrast: This slider let's you adjust the contrast ratio of the image captured and increase/decrease it.

A 50% value is the default state and indicates no changes to the Contrast. Slide right to increase the Contrast up to 100% (maximum contrast) and 0% to reduce it (lowest contrast).

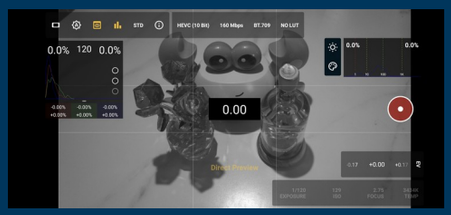


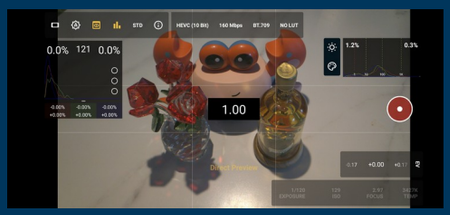


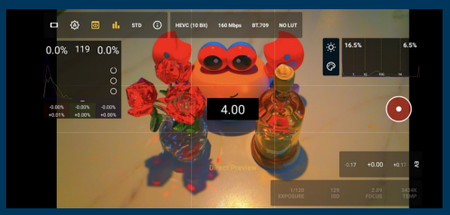


Saturation: This slider let's you control the image saturation rendered which will impact how intense colors look.

A 1.00 value is the default state and indicates no changes to the saturation as per the image matrix chosen. Slide right to increase the Saturation up to 4.00 (strongest saturation) and slide left to as little as 0.00 to decrease it (0.00 will become gray and retain no colors/becomes gray-scale).



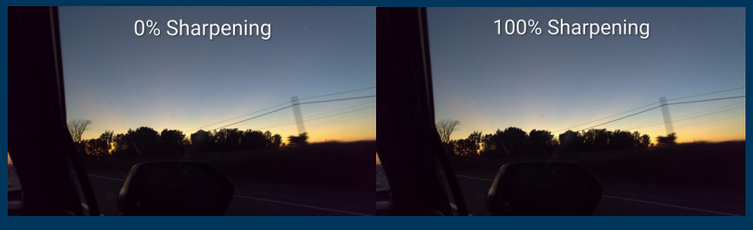


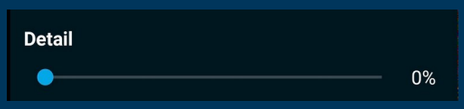


Sharpness: This slider let's you control the level of image sharpening applied to the rendered output

A 0% value is the default state and indicates no sharpen is being applied. Slide right to increase the sharpening level up to 100% (highest sharpening possible) and slide left to decrease it back. This setting will impact performance and requires more resources to apply it when enabled (±1% value).

Sharpening adds contrast around the edges of surfaces and textures in the image, creating the illusion of a sharper output. It looks for variances between pixels or sudden changes (which may indicate a different texture or object) and amplifies the contrast between them. Beware, however, as overdoing it will create an unnatural look with added noise and odd artifacts in edges, to name a few.

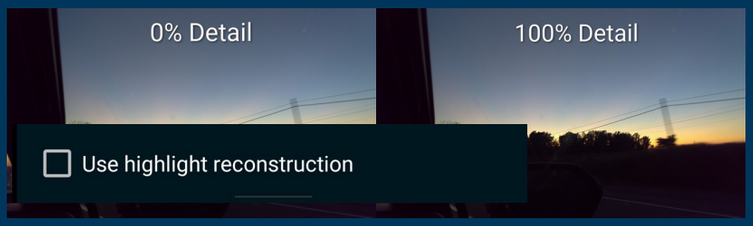


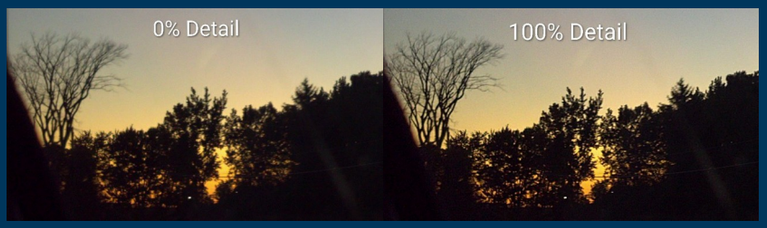


Detail: This slider let's you control the level of image Detail enhancement rendered which will impact sharp finer details look.

A 0% value is the default state and indicates no Detail enhancements are applied. Slide right to increase the Detail up to 100% (Maximum enhancement) and slide left to reduce it. This setting will impact performance and requires more resources to apply it when enabled (±1% value).

Detail enhancement, similarly to sharpening, attempts to also increase contrast between pixels that show variances, however it more so targets smaller surfaces and textures, providing with a more granular sharpening control that is less aggressive on bigger textures and surfaces but may amplify noise. Best used to enhance finer textures like landscape details and such. Use with care as too little will result in only larger surfaces being sharper and insufficient micro detail enhancement, too much will result in significantly amplify grain and produce artifacts.



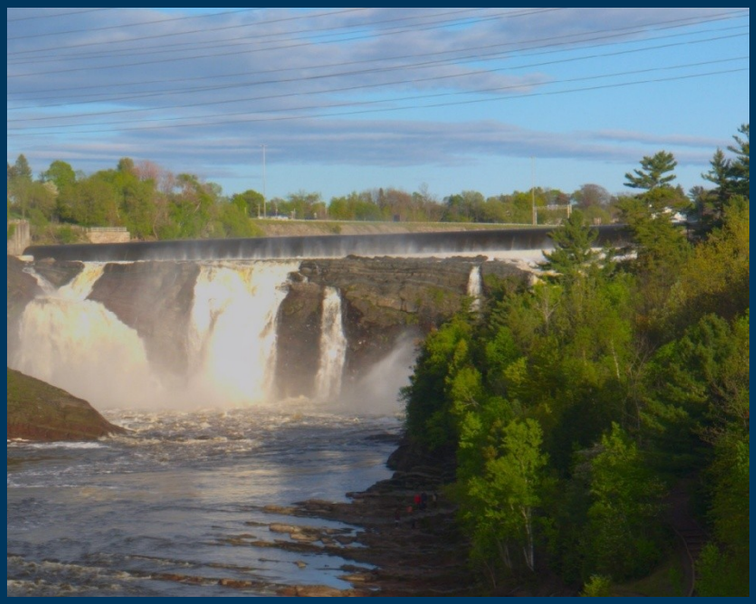


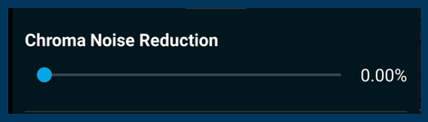
Highlight Reconstruction: This feature allows you to potentially reconstruct or recover clipped highlights by leveraging data from the color channels that have not yet been fully blown out. Warning: This option will heavily impact performance. Check the box to toggle On.

We will cover the in-depth use of this option later (see page ##), but the principle is simple: you intentionally use the Histogram to controllably clip one or two channels (Red, Green, or Blue). You then use the remaining, non-clipped channels' data to mathematically generate the missing information for the clipped channels, un-hiding the area that would otherwise show pure white.

By leveraging this additional dynamic range, you can push exposure further into otherwise unusable ranges to reduce noise and enhance shadow data. This technique is best used for specular highlights or surface lights. The reduced color accuracy from using incomplete color channels will impact color fidelity; in other words, do not use this to recover blown highlights on people/skin tones or anything that requires good color reproduction—otherwise, they will possibly look... interesting...

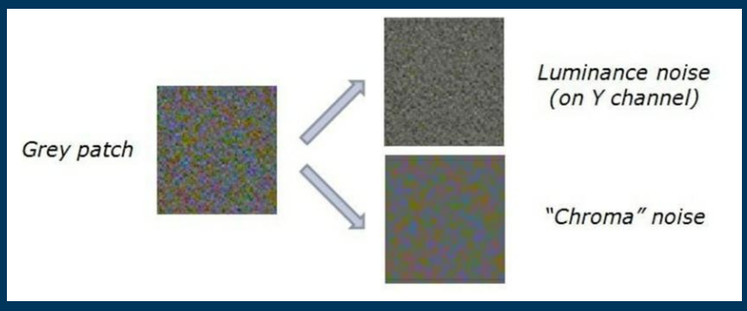
Additionally, although we'll cover them it later as mentioned above, MotionCam offers visual overlays to effectively indicate clipping areas and the amount of channels clipped, outside of the histogram alone. This makes this option quite a powerful tool, particularly for ETTR enthusiasts!





Chroma Noise Reduction: This slider let's you control the level of Chroma Noise Reduction applied on the rendered video data. This setting will slightly tax performance but can provide basic denoising.

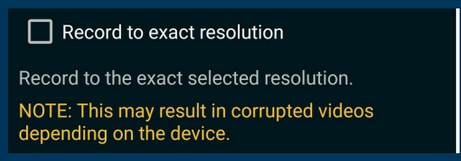
A 0% value is the default state and indicates no Chroma Noise Reduction is active. Slide right to increase in 12.5% increments up to 100% (Maximum out of 9 total steps) and slide left to reduce it. This setting will impact performance and requires more resources to apply it when enabled (±1% value).



Sensor image noise can generally be broken down in two categories, the Chromatic noise (aka – Chroma or Color Noise) and Luminance Noise (aka – Luma noise or image grain). This Setting attempts to isolate and target the random Chroma Noise which is the more visually damaging type if left unchecked. This random noise or color data provides some useful data in its variance however it can often become overwhelming in noisy environments and can start detracting from the image. There's multiple noise reduction methods, however this specific one attempts to target the random chroma noise in every single individual frame and attempts to mitigate it.

Color denoising comes with a slight risk of suppressing the colors particularly in micro details, however can also help to produce more usable low-light content. Although adding more light should always be the first noise reduction method, when that's not available, this makes a great tool without impacting performance drastically.

Advanced Settings



Record to Exact Resolution: Sometimes, your selected resolution might be slightly padded with extra pixels to ensure the container resolution is a multiple of 64. This is not a random effect! As the video is being processed, the device hardware often requires the resolution to be divisible by 64. If it isn't, the encoder can't work correctly, potentially leading to corrupted outputs, artifacts, or video anomalies (and you don't want that).

When this setting is disabled/unchecked, the state, the app takes care of this by automatically adding a few extra pixels on the border areas as needed, respecting the hardware's rule. We aren't going to touch on the deep programming logic behind this issue here; just know that the easiest path is to respect the rule and allow the padding.

For example, if you shoot in 3840x2160 (4K UHD), you might expect an output of something like 3840x2176 instead. This will vary on your specific resolution crop.

As for how this extra padding shows up: you can generally observe it right on the edge of your video frame, usually as a very minor mirror effect of one of the border areas. You can simply crop this off in post (or ignore it if it's minimal! How noticeable the mirroring looks will depend completely on your resolution and device).

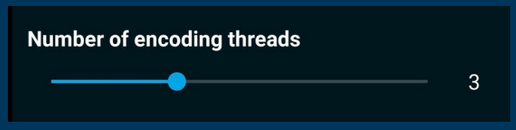
Enable this option to force encoding with the precise resolution crop you chose, skipping the padding. This can be great for workflow since rendering with the pixel padding isn't always necessary and all devices react differently. That being said, testing beforehand is essential, as forcing an exact resolution can potentially break, corrupt or create anomalies on your output! Feel free to experiment!

Buffer Size (Direct Log only): This setting allocates the size of the RAM buffer the app will use for capturing Direct Log video. Slide right to increase and left to decrease.

When capturing, the RAW frame data is placed into a memory queue (a waiting line) before the device's encoder can process it. An encoding deficit occurs when the incoming data overwhelms the device, meaning the queue builds up faster than the encoder can clear it. Once the queue exceeds the allocated buffer size, the data spills over, and the app can no longer hold new frames reliably, resulting in dropped frames.

Increasing this memory buffer provides the app with more room to hold the data, mitigating dropped frames for shorter performance bursts. It also gives you more reaction time when the device overheats and slows down, as you'll observe the memory usage filling up more gradually rather than being instantly overwhelmed.

You should use a higher setting if your device has sufficient RAM, though it is not mandatory. Be aware that if the size is pushed too high and the device has insufficient total memory, it may lead to app instabilities.



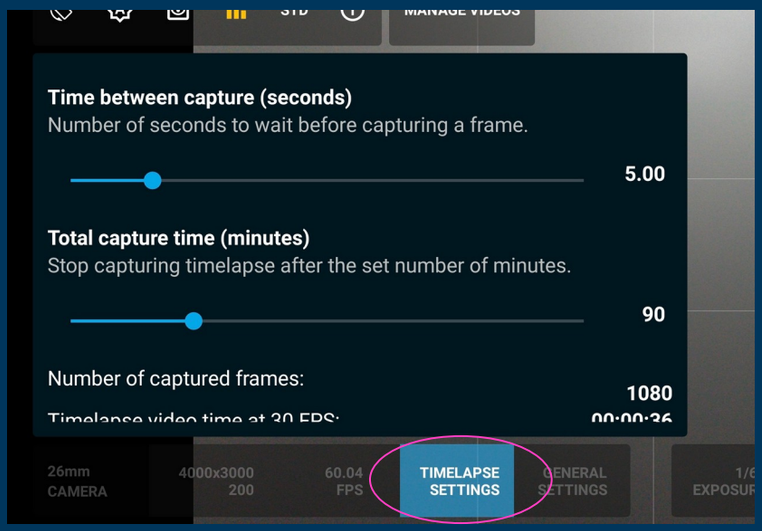
Number of Encoding Threads (Software Encoding): This setting lets you control the number of CPU threads used when encoding a codec that relies on the CPU for software acceleration (e.g., ProRes). This helps

optimize the data load on the processing side. Slide right to increase and slide left to decrease the count.

Note: More isn't always better! Modern CPUs often use a mix of powerful Performance (Big) cores and less powerful Efficiency (Little) cores. Depending on your device's CPU architecture and how it groups these cores, engaging too many threads might accidentally slow you down by involving the slower Little cores meant for power efficiency in the intense encoding task. Your mileage may vary significantly therefore!

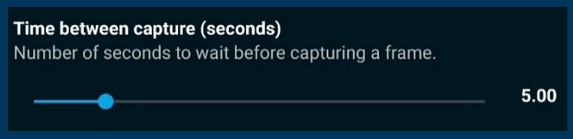
We strongly recommend experimentation. You might find better performance by only engaging a lower number of threads (to focus the work on the high-speed Big cores) rather than maxing out the setting and engaging every core, Big and Little, regardless of suitability for the task.

TIMELAPSE SETTINGS MENU



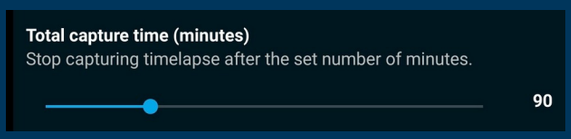
Please note that the next following UI Elements are only related to the Timelapse Mode

To access them, select the TIMELAPSE recording mode and select on the TIMELAPSE SETTINGS button on the bottom left, this will open the capture parameters menu for your timelapse session!



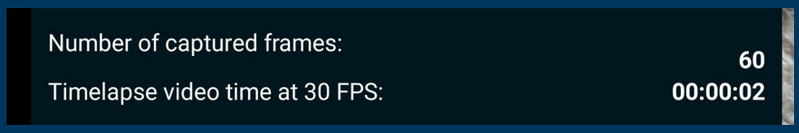
Time Between Capture (Seconds): This slider allows you to define the number of seconds/intervals that the app will wait for to space out the saved frames in the session. Any frames captured in-between the gap selected will be discarded to save space.

This value will provide for selections between 0.25 seconds to 30 seconds. Slide left to decrease and slide right to increase. The selected interval will appear to the right with quarter second precision.



Total Capture Time (Minutes): This slider allows you to define the duration of the timelapse session in minutes. After the selected time lapses, the app will stop Recording.

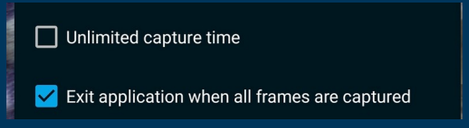
The selection can be adjusted in 5 minute intervals, with the shortest capture time being 5 mintues and the longest being 360 minutes (or 6 hours). If you require longer times, it is possible to override it with one of the following settings below, however (Unlimited Capture Time checkbox)



Number of Captured Frames: The value to the right can be used to indicate the amount of frames the app expects to capture assuming appropriate shutter speeds and framerates are selected (that will respect the exact timelapse settings of choice). The following calculation is used…

(60 Seconds ÷ Time Between Captures in seconds) * Total Capture Time in Minutes = Number of Captured Frames

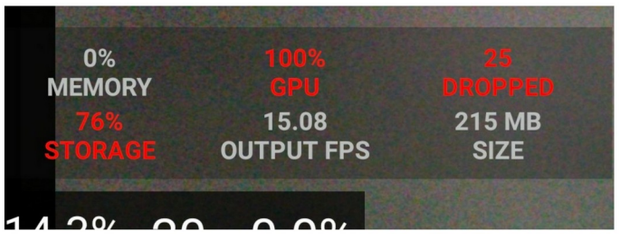
Timelapse Video Time at 30 FPS: This timer will indicate how long a video playback would last for the the above timelapse settings chosen, assuming the sequence was viewed at 30 FPS. Use it to determine if the session is sufficient or too short, although it's only a guideline.

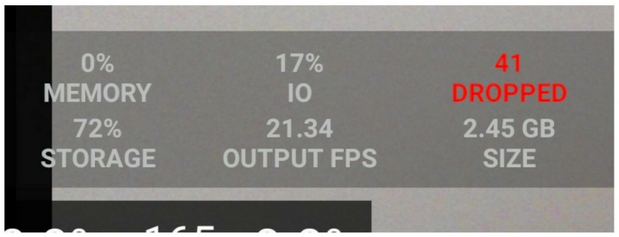


Unlimited Capture Time: Check this box in order to override the above Total Capture Time selection and force the app to capture indefinitely and override all related sessions duration settings. Using this setting will make the app no longer provide accurate estimates for the timelapse however.

Exit Application When All Frames Captured: This setting will make the app exit upon recording all the frames as per the above Number of Captured Frames calculation estimate. This is a great way to automate the session and allow it to end autonomously, however it may also be overridden with the Unlimited Capture Time option.

Recording Session Performance Interface

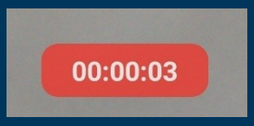




This following section will dive into the Recording Session Performance Interface, or otherwise, the section that appears when recording begins and indicates details including the storage used, frame drops, processing bandwidth available, and more.

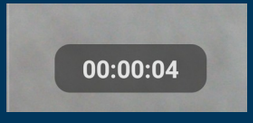
The following elements will only appear once a capture/recording session has begun and will not appear otherwise. Use them to evaluate how your device is handling the requested capture parameters.

Recording Duration Timer: When you begin recording, a session time counter will appear in the center upper region of the display. It will change depending on the situation to inform you of the current state and length of the session. Will appear in HH:MM:SS (Hours:Minutes:Seconds) format.

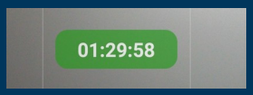


Normal Mode Timer: Shows how long you've recorded in the current session.

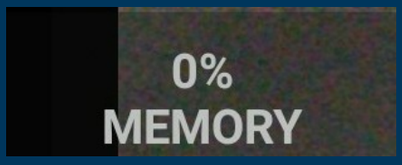
Buffer Mode Timer: Shows how long you've recorded in the current session and indicates Buffer Mode is in use.



Paused Timer: Will show the timer in a grayed out status whenever you use the pause button to indicate the session is paused. Will remain static until you press resume and it will then return back into the Normal Mode.



Timelapse Countdown: The normal timer will switch to green and into a countdown mode instead to indicate you are now shooting in timelapse mode and the time left until the session ends. It may behave like the Normal Mode timer however, if you selected the Unlimited Capture Time setting.

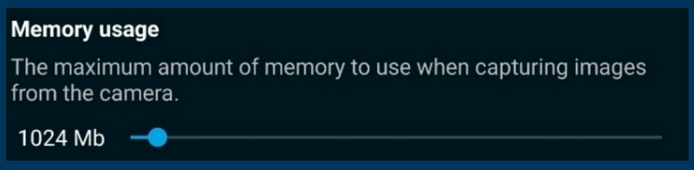


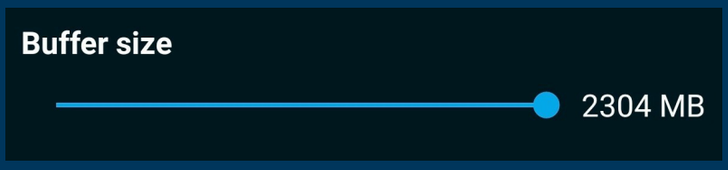
Memory Usage Percentage: This value displays the live usage of the RAM buffer allocated to the app. During RAW video or Direct Log recording, the incoming RAW sensor data is first placed on this memory buffer before the CPU or GPU can process it into the final format (like MCRAW or a Direct Log codec).

You can use this value to instantly assess your device's encoding efficiency. If the value sits near 0%, your device is successfully processing the RAW data faster than it comes in—you're stable! If the value starts climbing, however, it signals that the performance required exceeds what your device can output; the buffer is building up, and the device will eventually become overwhelmed.

When this value hits ±50%, it will turn red - this is your critical alert for an imminent Memory Overrun. This is your immediate signal to stop recording before your session is suddenly bombarded by frame drops.

While performance surges and drops are normal, this buffer provides crucial breathing room to mitigate drops in such event. If the cause for the performance bottleneck isn't controlled, the Memory will eventually reach 100%, resulting in frame drops. Use the Memory Buffer setting (pages ## & ##) to manage this allocation for better stability or performance.

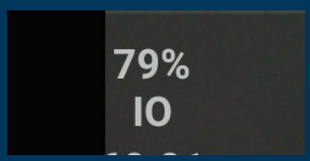




Storage Amount Used: This value displays the percentage of storage currently used in your device's selected recording location (Internal Storage or an External SSD). A 0% value means the location is empty, while 100% means you have no room left to record.

The value will turn red at ±76% to alert you that storage usage is reaching high levels.

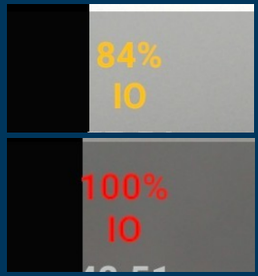
It is strongly discouraged to fill up any device above 85%. Doing so often results in a performance impact and can cause storage writing speeds/bottlenecks as the system begins throttling down the write speed. This risk varies by capacity; for example, 20% remaining on a 1TB device is far less critical than 20% left on a 128GB model. Never fill your internal storage to 100%, as this can create file management anomalies and severely slow down your device or even crash the Android system OS.



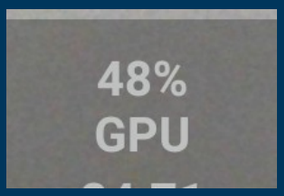
Input/Output Capacity (RAW Video Mode Only) : This value displays the percentage of available IO (Input/Output) currently used by the recording session. IO is your storage's ability to manage the flow of incoming RAW data against its ability to write that data out (as MCRAW).

A 0% value indicates no IO is in use, while 100% means you've maxed out your storage's writing speed, whether due to slow speed, throttling, or other bottlenecks. This percentage is a real-time gauge of your device's storage performance and session health. Lower IO values are always better, but as long as you can hold a stable value below 100%, performance is considered sufficient.

The value provides critical warnings: it will turn yellow for moderately high usage (80% to 89%) and turn red for critical IO levels (90% to 100%). Reaching 100% will begin overflowing data into the backup Memory Buffer and result in massive frame drops once that gets overwhelmed too.



Note: This setting will only appear during RAW video capture. Direct Log Video will show GPU percentage performance instead.

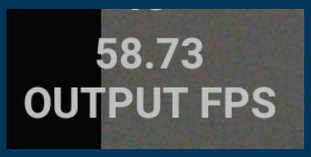


GPU Percentage Usage (Direct Log Video Only): This value displays the percentage of available GPU power being consumed during Direct Log Video captures, allowing you to gauge the performance demand on your specific device. You ideally want to keep this percentage as low as possible for best stability, but it will naturally fluctuate based on the scene, lighting, codec, container, and selected rendering settings (which have a big impact).

A higher GPU usage not only lowers potential stability but also generates significantly more heat. The percentage shown will adapt to reflect the system resources available, as thermal throttling itself reduces the GPU's available power.

A 0% value indicates no GPU usage or that the device isn't rendering. A 100\% value indicates you've exceeded the available processing power, forcing the device to immediately fall back on the recording buffer to absorb the shortfall. This will result in immediate frame drops once you exceed the buffer allocation. The percentage turns yellow from 81-90% to indicate higher than desirable consumption, and red from 91-100% to signal that you are about to, or have already, maxed out your processing power.

To summarize, lower is generally better, but this will be highly dependent on your device's power, actual performance and stability. You should also balance the usage against additional rendering capabilities that may be desirable to increase overall quality.



Output FPS/Frames Per Second: This indicator allows you to see how many image frames the app is able to capture and process into a container based on your chosen settings. Ideally, you want this number to be as close as possible to the framerate you've chosen for capturing. Use this value to assess the actual rendering performance the device is providing.

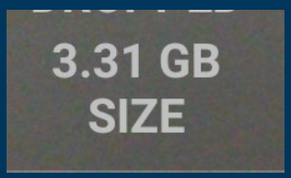
The actual output fps is generally going to be slightly lower than the framerate which is normal, however it can also drop way below depending on actual shutter speed in use, as well as based

on the rendering intensity. Very demanding settings that push the device beyond its limits will result in a drastically lower output, which will result in frame drops

Dropped Frames Counter: This area displays the total number of dropped frames within the current capture session. Dropped frames occur when the frame or image data given by the device cannot be successfully captured, resulting in a "gap" within the frame sequence. This failure manifests as a stutter or jaggedness in your final video. A low amount of drops can be tolerable, but a high amount will severely impact viewing quality and discard large amounts of incoming image data, making it highly undesirable.

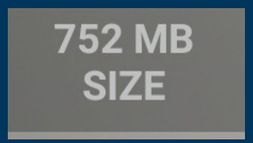
This indicator will show "-" to indicate no drops have been detected. Any value from 1-10 will show in white to indicate nominal performance, or otherwise an acceptable level of frame drops within nominal variance. Any values from 11-20 will show in yellow to alert you about a rising number of drops that may begin to become noticeable. Finally, any number over 21 will be shown in red to indicate a possibly critical level of frame drops.

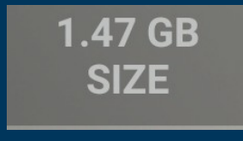
Ultimately, you will be the final judge of what is acceptable or not. For instance, a very long recording with widely spread-out drops will be far less noticeable than one which is shorter and has a large cluster of dropped events. This indicator is only a simple, quantitative way to measure them.



Real-time File Size : This indicator provides you with the real-time size of the file you've accrued during the current capture session.

The size indicator measures file size in Megabytes (MB), typically from 0 up to 1023MB. Once the file size exceeds 1023MB, the measurement automatically switches to Gigabytes (GB), rounded to the nearest hundredth decimal place (e.g., 1.02GB).



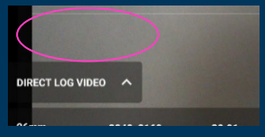


This value will generally keep climbing as you record. If the size indicator remains static while you are actively capturing, it signals a problem preventing the data from being written (a rare event).

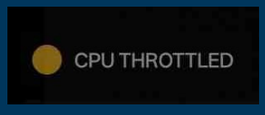
CPU Throttling Indicator: This thermal status indicator, located at the bottom left, provides a real-time assessment of your device's thermal/heating state and will appear regardless of whether you are actively recording.

The app receives alerts directly from the Android system regarding the thermal state, which is handled according to your Device Manufacturer's defined specifications on your specific device. While some devices throttle more aggressively than others, this indicator alerts you to specific temperature thresholds and the corresponding power reduction to expect. In optimal conditions, the app is allowed to utilize the full peak power of the CPU. However, as the device heats up, the Android system begins to moderate this available power at specific, undescribed thresholds defined internally as THERMAL_STATUS_######.

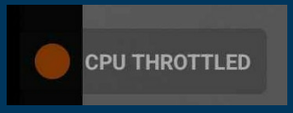
Just because you see this alert pop up doesn't mean you must stop recording; it simply tells you that your available peak performance power is being reduced. If your current capture settings require performance within the now unavailable power envelope, your session may no longer be sustainable. If your session doesn't require the full capacity, this is merely a heads-up that your headroom is shrinking. The app will warn you accordingly of the current thermal state with four distinct thermal states:



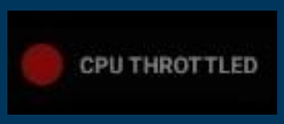
No Alert: Defined by the Android status THERMAL _STATUS_NONE, this shows the System is not under throttling and full power/performance is available for capturing.



Yellow Throttling Alert: Light throttling where Android UX is not impacted, however peak capabilities are mildly reduced. Mostly harmless however indicates heat buildup is beginning to increase. THERMAL_STATUS_LIGHT

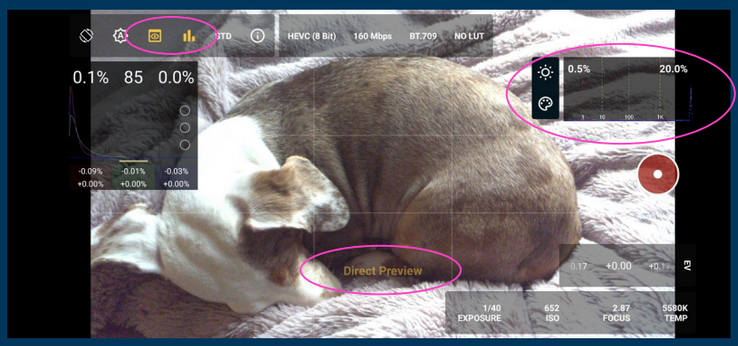


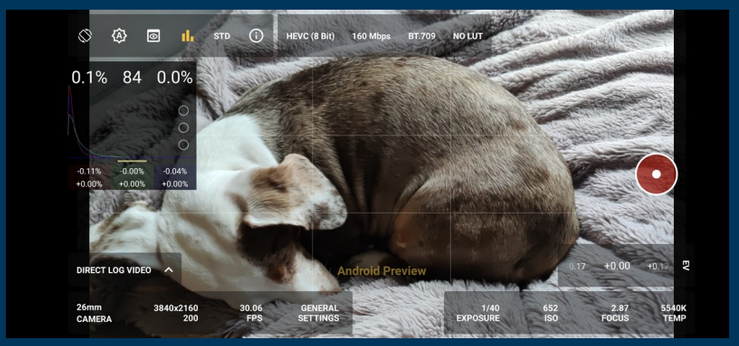
Orange Throttling Alert: Moderate throttling where Android UX is not largely impacted but apps like MotionCam in this case will begin suffering due to heavier processing restrictions. Your device will be noticeably warm and you'll begin to notice potentially less stability and more frequent frame drops. THERMAL_STATUS_MODERATE



Red Throttling Alert: Congratulation! You dun' did it now – you're cooking! This is the last indicator showing severe throttling where a massive performance reduction will be seen across the board. Expect your capturing capabilities to go down the toilet at this point. You should attempt to reduce strain or cool down the device if possible since otherwise, if you keep pushing , the Android system itself may begin to intervene further (may put the device on limp mode, shut down the device or begin throwing OS warnings). THERMAL_STATUS_SEVERE.

Direct Preview (Usage and Interface).





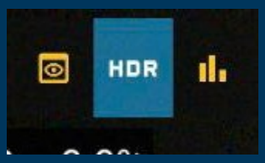
Upon pressing and activating the Direct Preview Mode, the bottom left bar will hide and a secondary dedicated Encoder Histogram to the top right will appear. Two buttons beside it to toggle the alternative Direct Preview Instrument Modes will also present themselves (A Sun and Paintbrush)

A prompt will also appear to indicate the Direct Preview Mode.

To summarize it again, Direct Preview is the app's unique mode allowing you to see the actual image content you are about to capture without any of your device's Image Signal Processing Features interfering or giving you an inaccurate, baked image that the app intentionally avoids. Without direct preview, what you are seeing is closer to what the stock app or standard video settings of your device would shoot.

When you turn on Direct Preview however, the app proceeds to process and show you the settings you've chosen, exactly as they would be as per the RAW stream source alongside any other settings for rendering you've altered. This is one of the app's most powerful and unique abilities, allowing for unprecedented precision and interpretation of the capture data well before you even need to shoot.

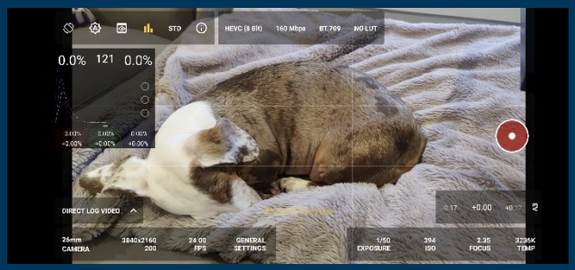
NOTE: You can also get to see your content using the HDR mode (if your device supports it and runs Android 14 or above), further elevating the capture previewing experience.

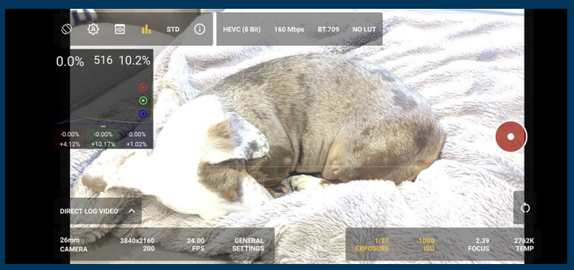


To compliment these capabilities, additional tools and modes leveraging the same control and purity of the source output are given to the user: a dedicated Encoder Histogram as well as False Color and Sensor Clipping overlays.

Direct Preview Modes: When you start the app or open a mode, a pop-up toast will appear to indicate what viewfinder mode is currently shown on the preview image. As of this release, you will have 5 modes. The below image columns will all be the same, with the left samples as normal exposure scenes and the right ones as overexposed samples for immediate comparison of the modes under different exposures.

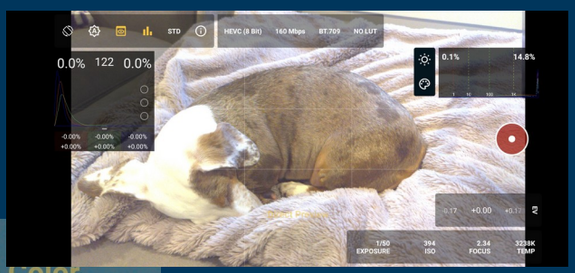
Android Preview: The standard viewfinder given by the Android system as per the Camera2API. The least resource intensive viewfinder mode, however will generally have processing applied to it. Although it's possible to diminish the processing applied towards the Android Preview (see page ##) it will still not generally provide you with a high fidelity/accurate preview and doing so may also show an image that's way darker or brighter than the actual capture – in some cases it may even crash the app due to system restrictions.

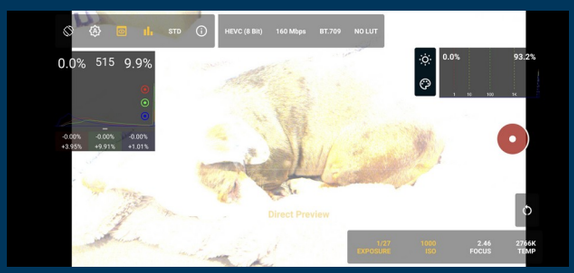




Direct Preview (Direct Log Mode): Defines the Viewfinder Direct Preview Mode is active with no additional overlays and will show an accurate Preview of your current Transfer Function (gamma/log) or LUT.

Direct Preview (sRGB/PQ) – Photo/RAW/Timelapse Video Modes: Will appear when capturing outside of Direct Log Mode. As the RAW data is technically linear, depending if whether you selected the preview to be in HDR or not, sRGB/PQ gammas will be applied respectively to the image to provide normal interpretation of the data as an idea of how it's being handled at the RAW level.

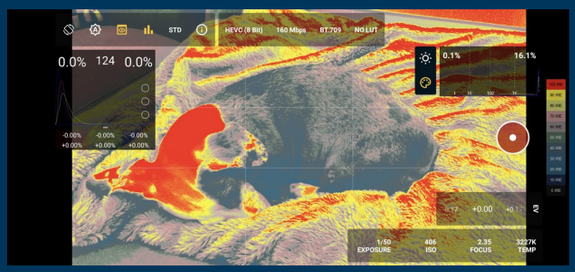


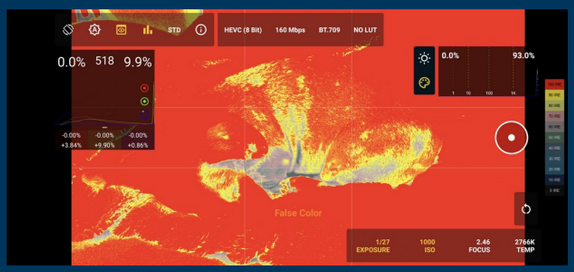


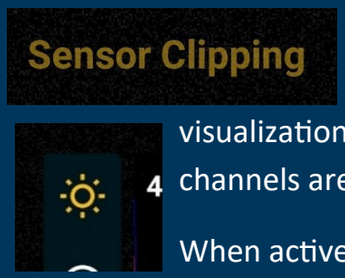
False Color: This pop up will appear when you press on the painting palette icon and indicates the False Color mode is engaged.

False color is an advanced exposure monitoring tool that emphasizes a scene's brightness levels by mapping them to specific, assigned colors. This visual system helps you quickly identify contrast ratios and analyze underexposed, properly exposed and overexposed areas, in a more pronounced manner, effectively ensuring correct exposure for recording is achievable regardless of the scene.

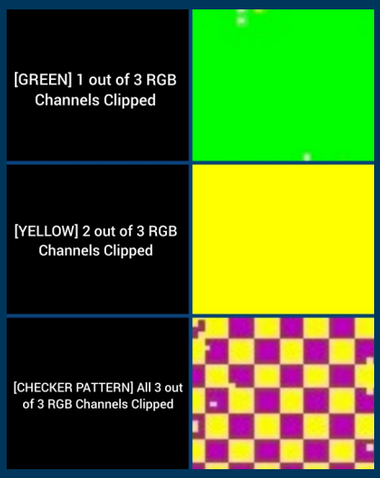
An additional IRE (Institute of Radio Engineers) scale will appear at the right hand side of the display to allow you to gauge the lighting and contrast ratios in a quantifiable and measurable way. A higher IRE value indicates a much brighter area and a lower one will represent deeper shadows until clipping eventually occurs.





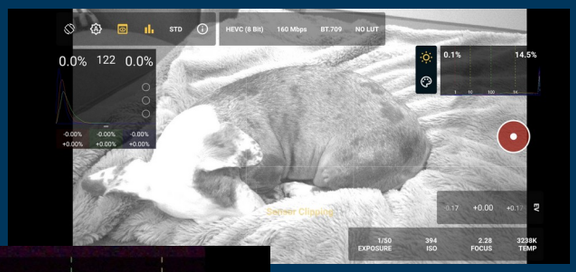


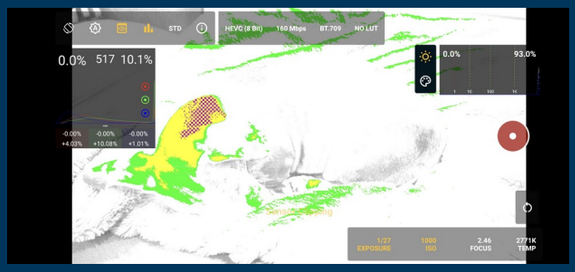
Sensor Clipping: This pop-up appears when you tap the shiny sun icon. This mode is unique to MotionCam. It leverages the app's special RAW data handling to give you a visualization of exactly where and when the individual separate RGB channels are clipping in the image.

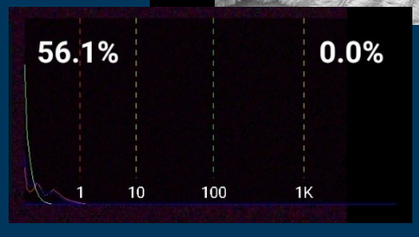


When active, the image appears in grayscale (black and white) to clearly accentuate light intensity, and when clipping begins, colored textures appear on the image. Green areas indicate where one channel has clipped, yellow areas show two channels have clipped, and a checker pattern tells you that complete three-channel clipping has occurred (undesirable, unless that area is not important to your scene).

This mode is designed ONLY to evaluate clipping of highlights and will not measure shadow clipping or crushing. Use this mode for precise and controlled ETTR (Expose To The Right) when paired with the Highlight Recovery feature - this method leverages partial sensor clipping (ideally one or two channels only) to help potentially extend your sensor's useful dynamic range by letting you eat further into the highlights! This can help drastically reduce noise as well.







Encoding

Histogram (Direct Log + Direct Preview Only) : When using Direct Log Video mode and activating the Direct Preview Mode, a secondary

Histogram will appear. This histogram's primary purpose is to indicate not how the sensor received light, but rather how that light will be packaged into your selected rendering settings (Transfer Function, Tonemapping, Contrast, etc.).

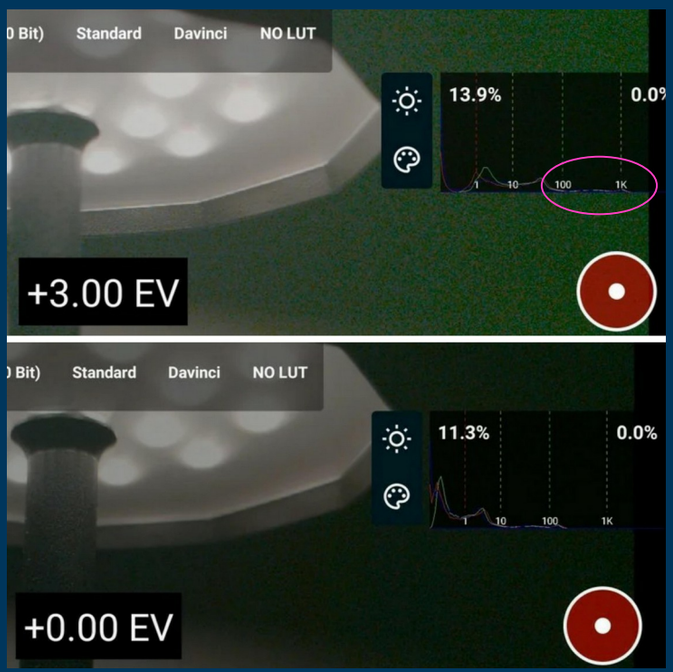
This histogram dynamically adapts based on your rendering choices. For instance, boosting the exposure gain will show the lighting increase accordingly, shifting both shadow and brightness details. Different transfer functions will also demonstrate different affinities in how the data is packaged and which areas of the container are utilized.

The two percentage indicators at the top are critical for measuring potential data loss:

•The top left percentage indicates the shadow clipped percentage—the amount of data expected to turn into absolute black after processing and will lose all detail.

•The top right percentage indicates the highlight clipped percentage—the amount of data expected to turn into absolute white after processing and will lose all detail.

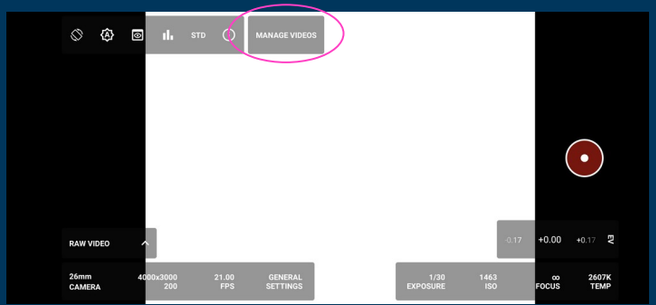
Any value higher than 00.0% in either indicator shows that clipping has occurred (right for highlights, left for shadows).

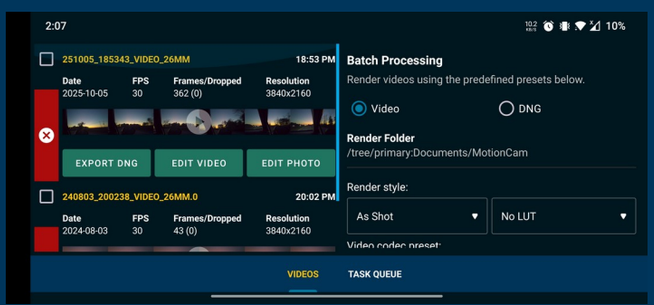


As is standard with Histograms, the Red, Green, and Blue lines indicate the amount and intensity of each color channel across the brightness range. The 1, 10, 100, and 1K (1,000) values along the bottom express the Nit levels (a measurement for light) of the scene.

To maximize container efficiency, you ideally want as much of the container used from left to right without any clipping. This histogram's purpose is to allow you to adjust your rendering settings to achieve this goal, giving you the ability to successfully measure the variance between how the sensor intakes light and how it ultimately compresses it. Harnessing this variance is the greatest power of this tool.

MANAGE VIDEOS (RAW/BURST/TIMELAPSE Only)





This following section will dive into the MCRAW Video Manager that comes built into the app. Here, you can manage, batch or individually prepare, process and render MCRAW reels into anything from single JPEG extractions, ready-to-go full CinemaDNG sequences, video codecs of your choice, and much more.

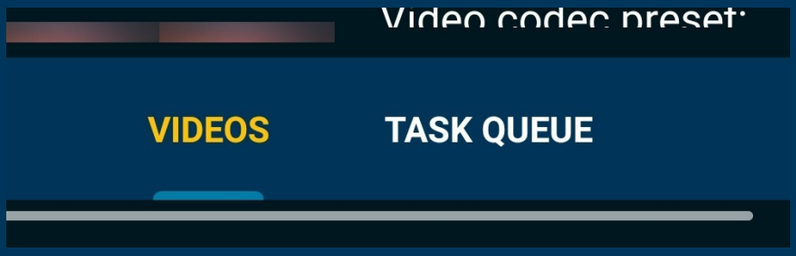
This menu is one of the most powerful app functions as you can effectively handle and process MCRAW format videos, regardless of which device shot them and without the need for an external device or PC.

It's important to note that although Direct Log videos share a similar pipeline to this exporter, their implementation is slightly different; that is to say, in some cases some devices will perform fine with this render but will not take to Direct Log (eg. Samsung Galaxy S20 FE). Additionally, more options such as denoising and additional codecs will be available here that may be too intensive to apply via Direct Log – although some settings are shared.

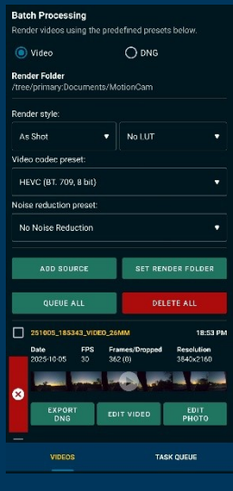
Additionally, this menu can work in Landscape or Portrait with identical functionalities available on both orientations.

This menu will be split into 2 sections, VIDEOS and TASK QUEUE. The active category will be indicated by Yellow writing as well as a blue underline.

Press on either mode to activate it respectively. VIDEOS will be where you can prepare and initiate the rendering process, TASK QUEUE is where you can monitor rendering time, progress, or finalized projects as well as modify when the app renders accordingly.

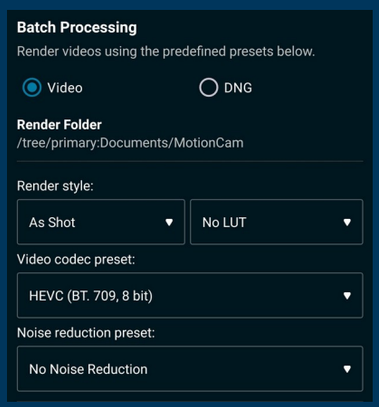


VIDEOS Tab (Video Manager): The default start-up mode whenever you open the Video Manager menu. Use this option to display any MCRAW reels found in your designated locations and to also export/render them!



You can scroll up/down to navigate it and explore all MCRAW videos shot with the app.

The app will provide you with the means to either process the reels individually, or as a batch. We will explore both approaches in the following UI element explanations as well as an additional step by step process will be available on page ##.



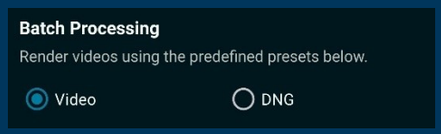
Batch Processing Interface: The top area of the Video Manager is designed to help you prepare and apply presets quickly and efficiently.

You can then apply these options in bulk to multiple MCRAW clips simultaneously to set them all in the rendering queue with your chosen configurations or adjustments!

It's important to note that the options you see here initially are strictly built-in presets included with the app to get you started.

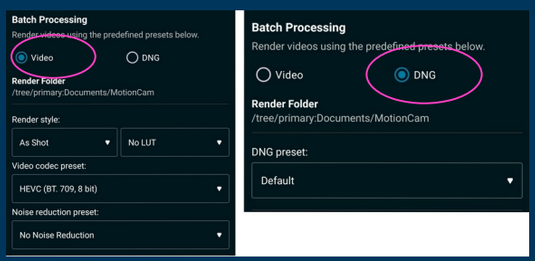
The power of this section, however, lies in your ability to add custom presets of your own! This includes custom combinations of codec variants and rendering options, such as granular denoising adjustments or even defining how DNG rendered files are interpreted.

Don't hesitate to move beyond the default options. Explore, experiment, and create presets that further suit your own specific needs as deemed necessary!

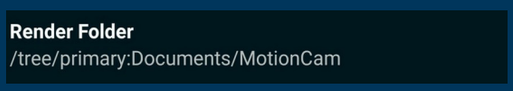


Batch Processing (Mode): This area indicates whether you've activated the Video or DNG batch preset modes.

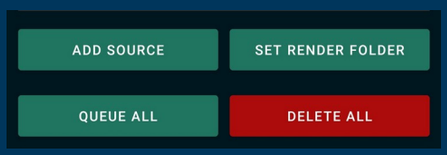
Press on either circle to switch and CinemaDNG sequence each bring up different indicate which mode is active.



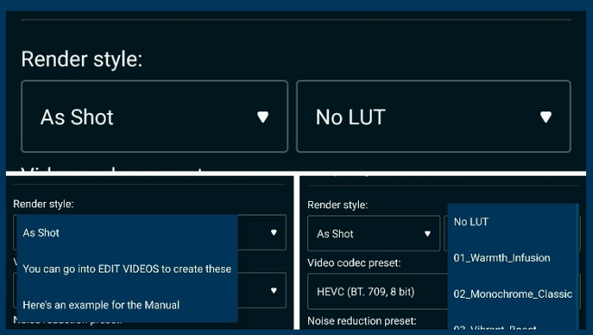
accordingly between video renderer rendering Preset options as they will options. A filled blue circle will



Render Folder (Location): This area indicates the file location in which current files the app renders will be saved on.



Can be adjusted via SET RENDER FOLDER option below the Batch Processing Menu area.



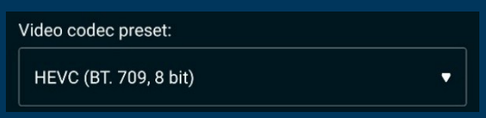
Render Style: Tap these box areas to quickly and efficiently select presets you've already created under the EDIT VIDEOS editor.

The left box (AS SHOT) is for in-app rendering settings you've altered and saved as presets.

The right box (NO LUT) allows you to select previously added LUTs to be baked-in as well.

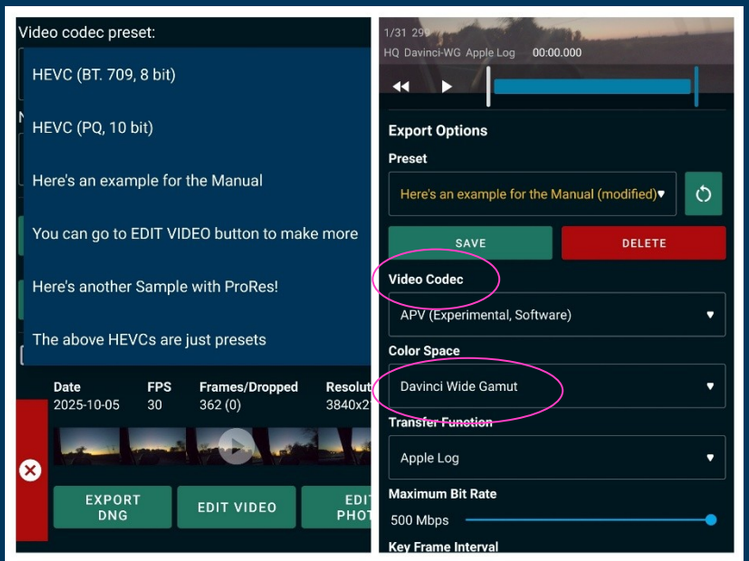
Both options can be used independently. However, keep in mind that if you apply only a LUT its transparency will be applied with 0% by default, meaning it's baked at full intensity and without blending.

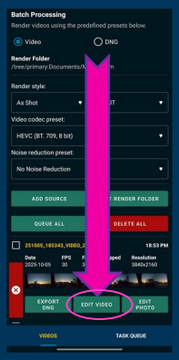
We will explore more on adding LUTs later on page ##.



Video Codec Presets: Use this box to open up the list of built-in app presets as well as the ones you've saved.

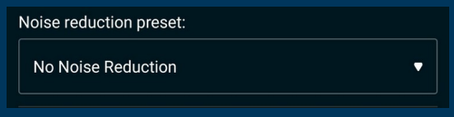
This list is often misunderstood and thought to be the only exporting options the app offers, but in reality they're actually just the preset names, not the codecs list itself! The current preset selected will be shown here. Tap on the box to expand it and see the full list and press on any other preset shown to use it instead.



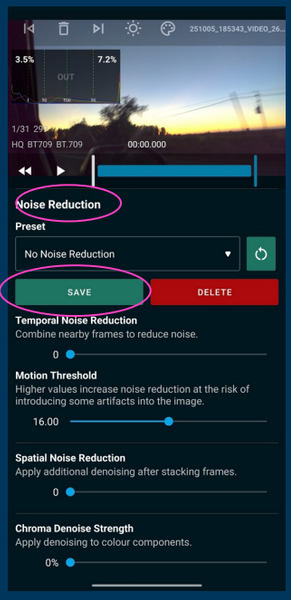


To create more, simply scroll down to view any MCRAW reel and press under any of their EDIT VIDEO option to open the Video Editor. You may then scroll down to the Export Options section to create your presets!

After changing the codec settings, simply press on SAVE and name your new preset!



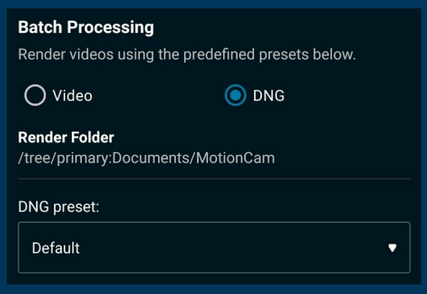
<u>Noise Reduction Preset</u>: Tap on this box to open the Noise Reduction presets you've already created. Just like the prior option for Video Codec Presets, these are only premade suggestions and you can create your own!



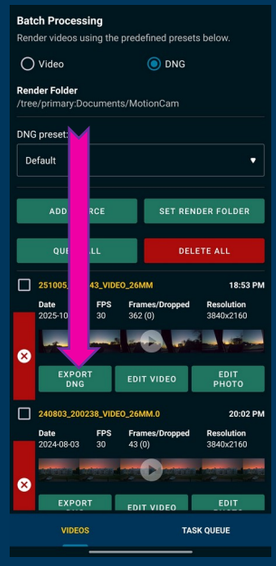
See the above Video Codec Presets steps to also create Presets — however in this case you are looking for the Noise Reduction section. Simply adjust the settings to taste and press on SAVE, then name your new preset!

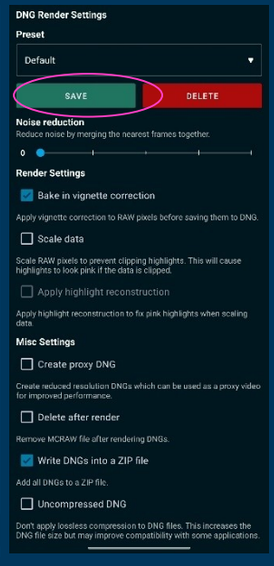
We will cover each Noise Reduction function later on in page ##.

BEWARE: Although using Chroma Denoise Strength is very efficient and quick to render, any usage of the Spatial or Temporal Noise Reduction settings will **significantly** affect exporting durations, so keep that in mind when deciding whether to apply denoising or not.



<u>DNG Preset (DNG Batch Processing Only)</u>: Use this box to quickly select any DNG presets for quick batch/bulk exporting of multiple MCRAW reels into CinemaDNG format.

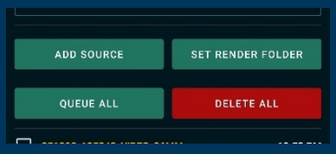




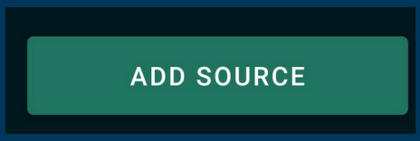
The standard presets are also mostly to apply denoising via frame stacking however you can once again create your own custom ones with further granular DNG data adjustments!

To create your own DNG Presets, simply scroll down to the MCRAW list and press EXPORT DNG under any of them; you will be brought into the DNG Render Settings menu. Simply adjust the settings and frame stacking

to taste, and press SAVE to name and store your new preset! We will further explore this DNG Renderer in further detail on page ##.

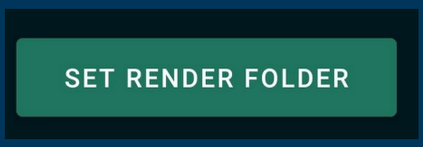


File Management: The following button cluster allows you to adjust settings in relation to where the app reads files from, renders them, as well as the batch handling of files.



ADD SOURCE: This option allows you to designate additional file locations (folders) that the app may look within to scan for MCRAW files (internal device storage or external locations such as SSDs).

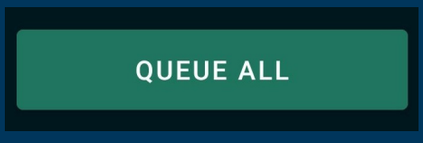
Pressing this option will open up your device's File Explorer. Simply navigate to your desired path of choice (must be a folder). **Note**: you cannot use the 'Downloads' parent folder due to Android restrictions however.



SET RENDER FOLDER: This option allows you to designate the file location under which the MotionCam app will save any rendered MCRAW outputs (regardless if rendered individually or in a batch).

Pressing this option will open up your device's File Explorer. Simply navigate to your desired path of choice (must be a folder). **Note**: you cannot use the 'Downloads' parent folder due to Android restrictions however.

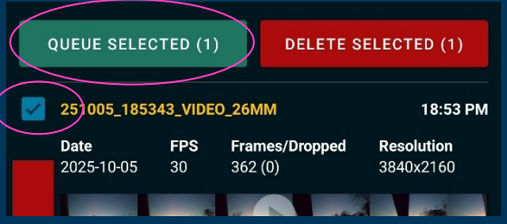
Although Internal storage is the safest option for this, you may also use external storage to save rendered outputs. Do beware however, if you've selected any external storage location that gets disconnected during rendering, this may result in the outputs being corrupted and the session itself will also be interrupted!

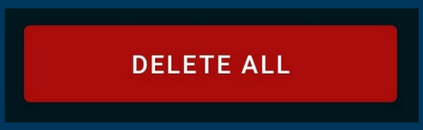


QUEUE ALL: This option can be used to apply the current Batch Processing presets as selected onto the full list of MCRAW reels that appear in the list below.

This setting should only be used as-is if you intend to batch render everything as it will select all the MCRAW files as found on your designated source locations and will begin a large Rendering Queue sequence.

As an additional function, you can designate which files are batched to be rendered by simply checking the desired reels boxes above them individually. The **QUEUE ALL** button will then change to **QUEUE SELECTED (# OF FILES SELECTED)** which allows you to batch only the selected group.

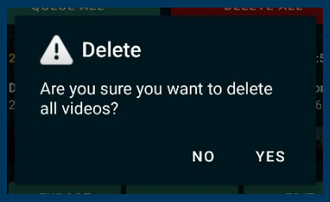




DELETE ALL: You see the red color? Do not press the red button… Ok, so actually this button is self-explanatory; if you press it, it will Delete All – yes, ALL of the MCRAW files that are currently shown per your designated sources.

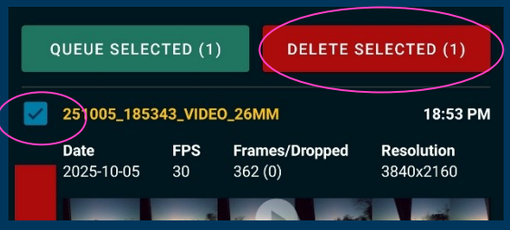
Only use this button if you've ensured that you no longer need any of the MCRAW files shown as once deleted, there's zero ways to recover the MCRAW files.

NOTE: This does not impact any existing rendered outputs, this option is strictly for MCRAW reels detected.



Great! You've pressed the red button… Well, you can rest assured you still have a fail-safe prompt in case you accidentally pressed it. You will be asked for a final confirmation of the deletion intent. Jokes aside, once you accept it by pressing YES, understand that the action finalizes and all MCRAWs will be permanently erased.

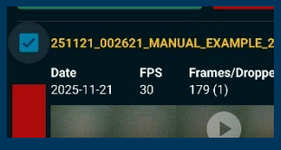
Lastly, this option will also change if you've selected specific MCRAW reels by checking their boxes. Simply select the desired group that you intend to delete and upon selection of them, the button will change to **DELETE SELECTED (# OF FILES SELECTED)** to indicate one or more files were individually selected.

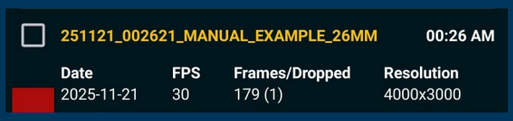


MCRAW WORKBENCH: Use this area in the Video Manager to preview and manage individual MCRAW files/reels found in your current file source location. Each file will be independently shown with its own Workbench allowing you to read details regarding the session and parameters at the time of recording.

You can also use the in-app tools on the lower area to export full CinemaDNG sequences or to open the built-in editors! We will explore these in the area below

You can also selectively batch MCRAWs by tapping and checking the top left box which enables the ability to select specific reels only.

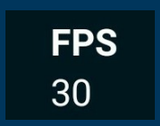




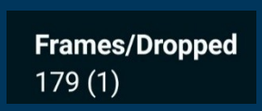
Session Review: Each MCRAW file will list the session details. The following information will be shown to quickly identify both the content and when it was shot, as well as some essential capture data.

The file name is the writing in yellow. To understand the app's naming scheme, see page ##. You can also observe the time the file was recorded at to the right of the name in white (AM/PM). You can also see the recording resolution listed in Width*Height values.

There are important things to note regarding FPS (Frames Per Second) and Frames/Dropped counters however.



The Framerate itself will ideally be what you've selected, however the app will round to the nearest stable framerate achieved and doesn't reflect the framerate slot selected, but rather the most stable one. If you've selected for example 60fps, assuming it runs fine but with many frame drops, you may for example see this counter report 59 FPS in such case. Additionally, using longer exposures that override the framerate may further deviate this value.

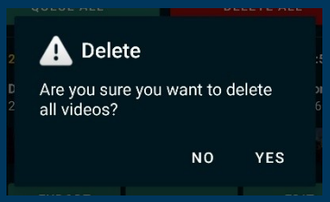


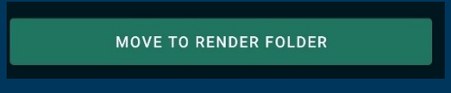
As per Frames/Dropped, the left number indicates how many image frames you captured in total during the session and the number in parenthesis indicates the amount of dropped frames – or in other words, the amount of frames that couldn't be retained and were completely lost. Dropped frames may occur due to a multitude of causes, such as performance or capture issues, bottlenecks , or thermal limits (potentially even misconfigured settings).

.

Single MCRAW Delete: We've already explained what the red DELETE buttons do in the Batch settings, however the red X button that shows to the left of every MCRAW reel allows you to specially delete the targeted file that's beside it only.

This will perhaps be the most used deletion method to work with. If you select it accidentally however, don't be alarmed, a final confirmation prompt to verify the intention will appear. All deleted MCRAWs are permanently wiped, so ensure you've reviewed or confirmed the file first since once you accept, there's no going back.

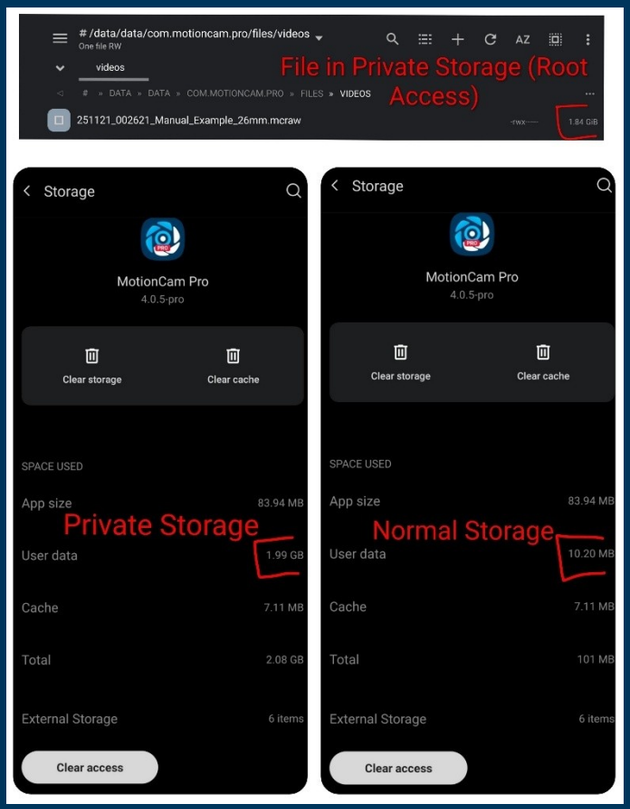




MOVE TO RENDER FOLDER: Press this option to transfer the MCRAW files out of the app's private storage and to bring into the ordinary designated render folder. Failure to do this may result in complete file loss if the app gets deleted.

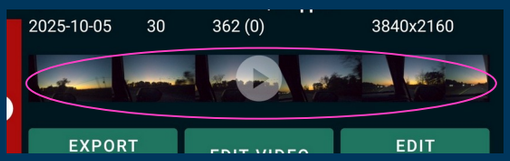
This area will ONLY appear if the app is currently storing any MCRAW files within the private storage area – no, we're not talking about degenerate scumbag folders; Android apps are allocated space within the device root folders, or otherwise, the area where the literal installation and program files are kept.

As discussed in page ## (***Capture to MotionCam Private Storage*** setting), the app can be made to instead write MCRAW file captures within this 'private' area at the system level, only accessible otherwise via root privileges and root explorers.



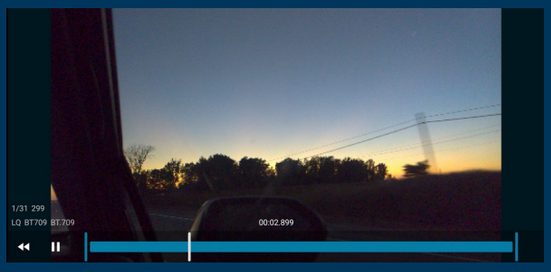
The benefit of using this option will depend per device, however in some cases it will provide faster storage I/O and eliminate system write speed bottlenecks. The downside is you then need to offload them using this button which is not an instant process

You can observe this by the app User Data consumption if you should use Private Storage; It will quickly make PUBG look like a Nintendo 64 file sizes game.



MCRAW Preview: This area provides you with 6 image frame previews from your sequence, split in equal intervals during the video to show you what was captured within the clip.

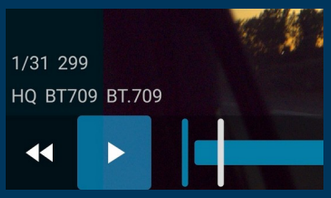
You can use this to identify the specific content at a glance, however you can also tap on the Play icon to open the in-app MCRAW Viewer!



In-App MCRAW Viewer: Upping tapping on the MCRAW preview area's play button, you will open the app's native viewer.

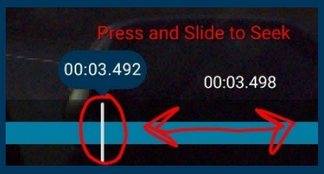
The files as seen here are not inherently at the maximum quality to ensure smooth playback (future app versions will solve this however this is the current mode).

Although they may not fully reproduce the quality or colors actually present in the files, this player will allow you to view MCRAW files as a normal video and play them back to assess them as needed. The preview will be with BT.709 color space and transfer functions both for displaying compatibility.



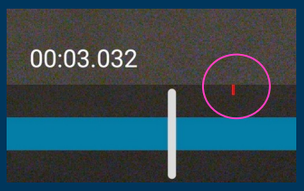
At the bottom left, you will have a ⏸ / ▶ button respectively to pause or play, and a rewind (⏪) button which you can use to fully rewind the playback of the file.

The top numbers row indicates the exposure/shutter speed to the left and the ISO reading to the right. The bottom row shows the color space and the transfer function applies to the preview, as well as the quality of the image shown (pause to increase to HQ and playing will decrease it to LQ).



You can also hold the white vertical line on the progress bar and slide it left/right to navigate and seek a special video moment.

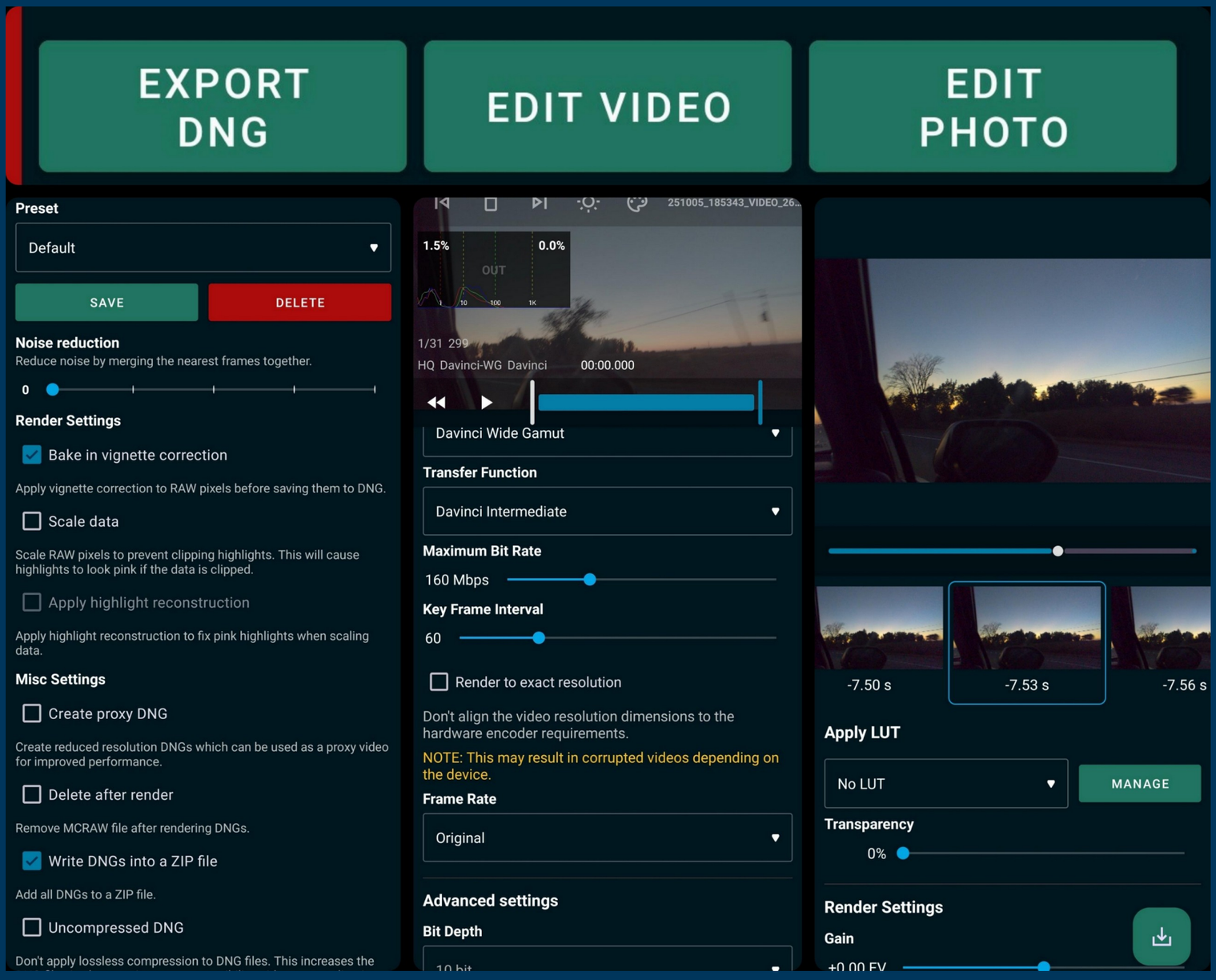
Additionally, the specific video time elapsed will show under white numbers and the current image being seeked when moving the white line will also be shown under a blue bubble to allow precise navigation of the footage.



Lastly, minor red indentations will also show above the progress bar to alert you of areas in which the app believes a frame drop occurred.

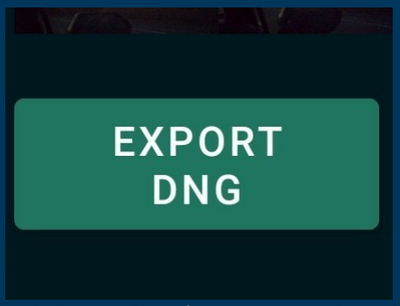
Although the MCRAW Workbench will advise of precisely how many, this tool will show you exactly where they occurred within the footage – a great tool to assess if the frame drops are going to be a concern or if they're spread out/clustered in a specific area/moment in the footage. As an example, if a bunch of drops occurred for any reason, but only during a section that you didn't need, then the file may still be completely usable.

IN-APP MCRAW RENDERING AND EDITING



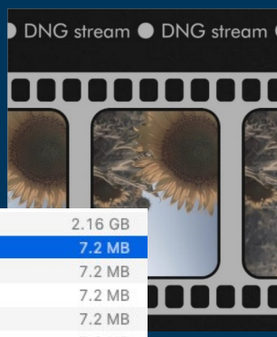
This next section will explore the often ignored, yet ultra powerful in-app rendering tools for DNGs, Videos and Photos (see their respective menus below them in the above example).

As you've probably learned through the manual, the app captures in .MCRAW format, a data lossless RAW video format. You can choose to offload it for processing at a PC, however if you also prefer a mobile workflow or want to leverage the powerful app rendering engine, the following buttons will provide you with all the tools required to process MCRAWs locally!

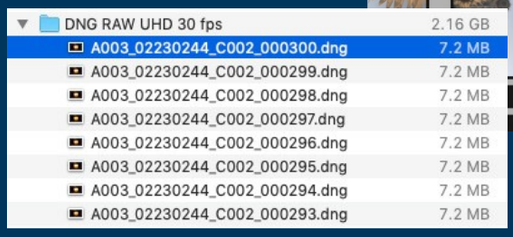


EXPORT DNG (DNG Render Settings): Use this menu to individually select a MCRAW sequence and the settings it will use before exporting it into a DNG sequence. You can also use this menu to create presets for batch DNG rendering.

Although it's not explicitly stated, this button could be technically referred to as the 'CinemaDNG' Converter, since it exports a version of the MCRAW file's data into a classical cDNG sequence.



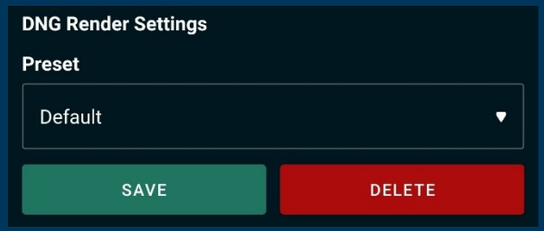
For the uninitiated, CinemaDNG (cDNG) is an open-source RAW video format that was created by Adobe. It consists of simply stitching together a sequence of DNG photos (Digital Negatives, also an open source RAW photo format created by Adobe) into a video, alongside with audio to go with. This essentially would look like a folder full of individual RAW images alongside an audio file.



Each image would be labeled in a sequential fashion that lets your editor understand which order they follow and this can then be played as a video with audio… RAW video! This really is like those classical film days, except digital!

Although widely supported, cDNG is rather inefficient as a format (as far as space usage and write/read speeds are concerned) and cannot compress anywhere near what MCRAW can. It also provides the added bottleneck of having to move over those folders which oftentimes contain hundreds/thousands of DNG files!

Nevertheless, if you want easy compatibility, this will be your route to use! The app's DNG sequence renderer above will give you the tools to simplify your workflow and modify/enhance the data pushed into the DNG containers; we'll explore that in the next few pages!

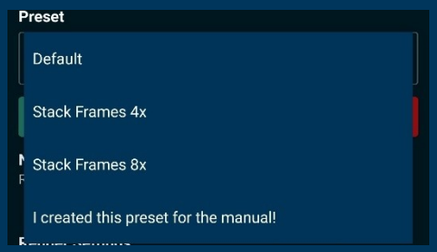


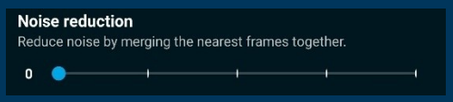
Preset (DNG Export): This setting allows you to quickly select or create rendering presets to be used for exporting/rendering DNG sequences.

Simply tap on the box to open the presets already created. Two existing frame stacking presets already come built-in, however you may add or delete more.

To save a new preset, simply ensure you've finalized all your settings in the rendering menu, then press the SAVE button above. You will then be prompted to name your new preset!

To delete a preset, simply select it and then press on the DELETE button on the red box.





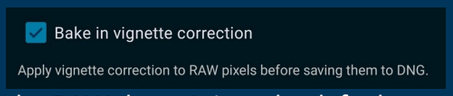
Noise Reduction (DNG Export): Use this option to apply noise reduction in the form of the app's signature Temporal Frame Denoising algorithm. In short, this option uses a custom algorithm that analyzes the image frame, then proceeds to use the data in the immediate frames before/after it to evaluate what is noise and what is not.

You can keep it at 0 for an unaltered DNG, and you can increase in increments of 4 by sliding to the right. A higher frame merge may potentially induce movement artifacts if the scene is too dynamic however. Additionally, this option will heavily impact rendering times; use as needed since not all scenes require it

What's that? You thought DNGs had to be pure and unaltered data? We've got something to tell you then..!

The advantage of denoising at the DNG level is in having the full RAW data from the frames before/after for denoising rather than compressed images which fundamentally have less precision. This algorithm is quite robust to movement and dates back to the MotionCam app's life as a Photo app only! Indeed, this is a relic of that era. A future upgrade is coming in newer versions however (far less resource intensive as well).

As a rule of thumb, to determine the effectiveness of Temporal Frame Merging, for every doubling of frames stacked, you reduce noise by half – this means it will require exponentially more frames to appreciably improve the effectiveness as you push higher.



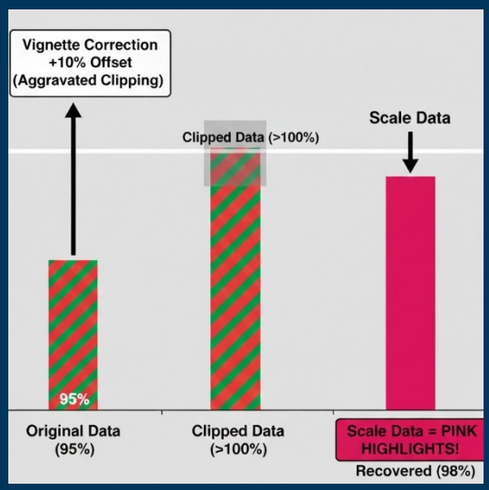
Bake in vignette correction (DNG Export): This setting allows you to irreversibly bake the vignette correction offset into the RAW data. It is on by default.

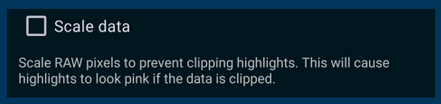
Ordinarily, this correction is bundled into the file as part of the EXIF metadata (for more details, see the vignette correction explanation on page ##).

Baking it in ensures you don't have to worry about the bundled data not being applied correctly by an external editor, or being stripped off accidentally. However, this process comes with two important trade-offs: you can no longer decide when you want to disable the correction if you've discarded the MCRAW, and you also reduce the effective dynamic range of the image, particularly near the edges of the frame. Let's explore why, and the solutions

NOTE – we will use percentages for simplicity's sake, but in reality we're talking about the white levels and their tonality values available to describe brightness. Vignette gradients are also complex things with all sorts of weird oval patterns and such. Offsets aren't merely perfect circles that get darker linearly //////

If your sensor reads a data value near the edge that pushes it—say, to 98% of its maximum dynamic range—and the vignette correction then adds an extra 10% boost, you've now digitally pushed the data beyond 100% of what the sensor could capture. This is a form of digital clipping caused by the added offset. This issue leads us directly to the next solution: scaling the data back…



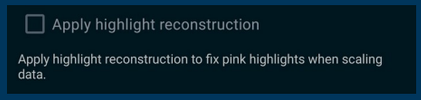


Scale Data (DNG Export): This setting only becomes available when you check the 'Bake in Vignette Correction' setting above. It allows you to forcibly scale down any data that may have clipped due to the vignette correction, and can also be used to further manage sensor dynamic range in extreme scenes.

As we've covered above, your dynamic range can be decreased because the offset correction digitally pushes data that was near clipping into the clipping range. Imagine some pixels are at 95% capacity, and a 15% correction boost pushes them to 110%. Your sensor will clip the data once the digital correction occurs and it touches any value ±100%.

Scaling back the data essentially forces the app to bring any clipped data back down if it observes clipping occurs after offsetting, before the correction finalizes and discards the data. To follow the example: if the app realizes the data was pushed beyond 100% after the correction, the data will be forcibly brought back to, say, 98% wherever clipping may have happened. While this is not quite the original value, it forcibly unclips the data so you don't completely lose it, thereby restoring some of the useful dynamic range otherwise lost.

Be aware that this option may potentially induce pink/magenta highlights for clipped areas. This is because if one channel clips completely, the remaining two unclipped channels are now unbalanced. For example, if the Green channel clips but Red and Blue remain, the area will be revealed as a magenta mix because the Red and Blue values now dominate and the necessary color information from the third channel is missing. This is a problem – and we like problems here. You can fix this with an advanced editor; or as you may have guessed, we've got something for that too; enter Highlight Reconstruction… (next page)



Apply Highlight Reconstruction (DNG Export): This setting only becomes available if you've activated 'Scale Data', which itself requires 'Bake in Vignette

Correction' options that were both mentioned prior to be enabled. Use this to algorithmically attempt to reconstruct the clipped highlight data whenever one or two color channels have experienced clipping.

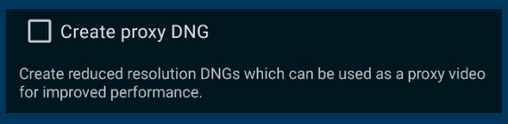
This feature is powerful because it completes the workflow. As we learned, forcing the app to Scale Data back (unclip) when a single channel is blown out creates that undesirable pink/magenta tint because the color is unbalanced. Applying Highlight Reconstruction steps in at this exact moment, using the remaining data from the two unclipped channels to mathematically generate the missing information for the third (clipped) channel. This effectively corrects the color balance and removes the magenta cast.

This ability to recover and correct clipped channels allows you to further push the useful dynamic range of the sensor. By using this trio of settings, you can intentionally expose further (e.g., utilizing ETTR—Expose To The Right) to reduce shadow noise, knowing that this powerful RAW pipeline will manage and restore the highlight detail and color fidelity, even if the clipping occurred due to normal scene conditions.

This highlight recovery method is applied directly at the DNG/RAW data level, distinguishing it from the video-level highlight recovery applied during the subsequent codec compression stage. It is best used to mitigate vignette correction losses unlike the video highlight reconstruction.

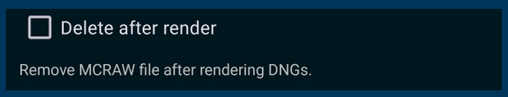
For more on highlight reconstruction you can jump back to page ##.

Misc Settings (DNG File Management):



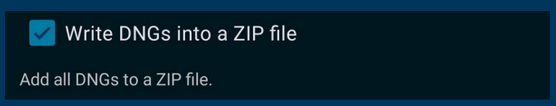
Create Proxy DNG: Toggle this setting to create 'proxy' versions of your rendered DNG sequence.

Proxy files are mini versions of reduced quality and resolution, intended to be used as temporary stand-in replacements for your main sequence files. This allows you to edit your footage with much lower processing requirements compared to using the full-quality files. Keep in mind this will take slightly more storage space, as you are creating a secondary, lightweight version of your reel specifically for editing efficiency.



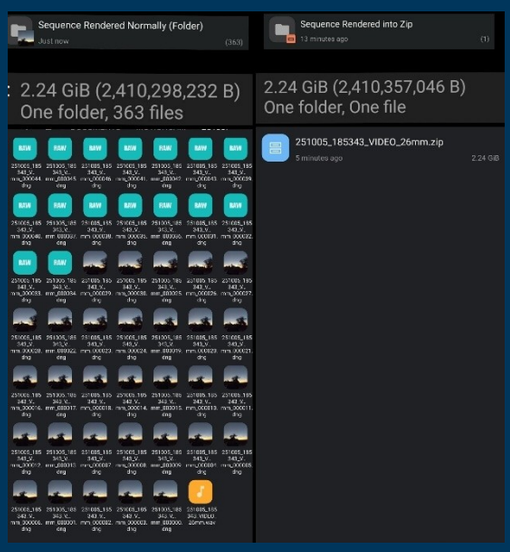
Delete After Render: Toggle this option to have the app automatically delete the original MCRAW file data upon successful completion of the DNG sequence rendering.

Ordinarily, the app's MCRAW files are retained after rendering, meaning you would have two RAW video versions of your capture. Use this option to immediately reclaim space by only keeping the converted CinemaDNG file outputs. Beware, however—this option irreversibly deletes the MCRAW file, meaning once you've rendered, there is no going back or altering the DNG rendering parameters (they become final).



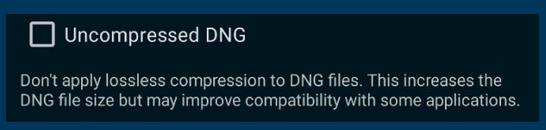
Write DNGs Into A Zip File: Use this setting to bundle your rendered DNG files into a single ZIP file to allow for quicker file management and drastically improve transfer speeds. On by default.

Normally, the files would appear in a folder containing hundreds or even thousands of individual DNG image files and an audio file, as shown in the left side of the accompanying image. While this method is straightforward for storage, moving around that many small files is significantly slower, due to the way computing devices manage IO (Input/Output), than simply moving one large file.



This option mitigates that inefficiency by outputting the entire sequence as a single, tightly packed ZIP file (as seen on the right side of the image). You can then simply transfer this single ZIP file and unzip it on your PC. While this adds the necessary step of unzipping, the time saved during the transfer due to avoiding the IO bottleneck of moving thousands of small files can be substantial.

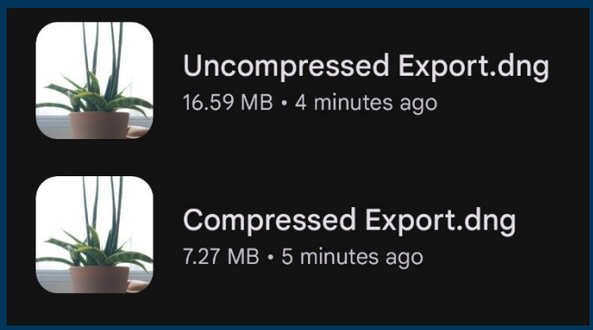
This option helps to mitigate the inherent clunky inefficiency of the CinemaDNG format, which we must use because most editors cannot directly work with MCRAW files (more on this later).



Uncompressed DNG*:* This option allows you to export DNG files with no data lossless compression applied. We do not generally suggest using this method, but it exists as a necessary auxiliary option.

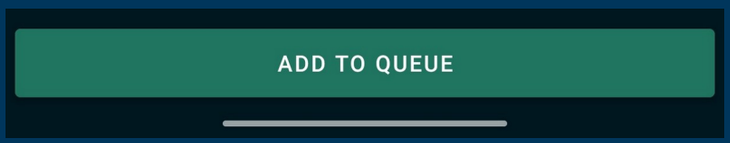
Here is why: Android devices often use DNG containers topped at 16-bit as the maximum (even if the sensor itself shoots at a lower depth, like 10-bit data within a 16-bit RAW_SENSOR container; see page ##). MotionCam utilizes data lossless compression to ensure only the actual data is retained. For example, if a 16-bit container only holds 10 bits of sensor data, the wasted 6 bits of empty padded space are efficiently discarded.

Furthermore, the compression algorithm can further shrink the file size lightly based on scene characteristics; the better the lighting and the lower the noise, the more effectively the compression can work. Although nowhere near as efficient as MCRAW compression, it nevertheless gives a noticeable space reduction.



However, some editing/processing software (looking at ya, Adobe Premiere) absolutely detest this type of intelligent compression and requires the full, untouched, and uncompressed container to function correctly. This is the only reason this option is available: in the unfortunate event that an issue like this occurs with your editor of choice, you can provide the fully uncompressed files instead.

To visualize the space savings: If the RAW data is ABCDEFGHIJ but is put into a padded 16-bit container ABCDEFGHIJ000000, MotionCam's compression saves space by trimming off the unused padding (000000). Uncompressed DNG forces the output to include that padding.



ADD TO QUEUE (EXPORT DNG): Once all the settings and/or presets have been selected, press this button to send the MCRAW file into the Task Queue for rendering.

Pressing this button will immediately return you to the prior Video Manager menu. You can then monitor the rendering progress and time remaining of the individual reel task (jump to page ## for the general explanation of the task menu).