OptArc AF51 Camera

User Guide

Document created: 28.10.2022,   Last edited: 23.05.2023

Manual prepared by OptArc – Designers and manufacturers of custom optical and optoelectronic systems using 3D printing rapid prototyping and based in the UK. See optarc.co.uk for details.

Safety Information

Identification of risk

Throughout this manual please take heed of **warnings given in bold text and highlighted yellow** to avoid possible damage to equipment and/or harm to people.

Risk to vulnerable groups

OptArc lab supplies and equipment are not toys. They contain small parts which may come loose, glass components that may splinter or break or otherwise present a choking or sharp object hazard. **Please do not let babies or young children play with or gain access to any aspect of this product. Older children should only access this product with close appropriate adult supervision. Likewise keep this product away from pets**.

Risk of damage to eyepiece

This eyepiece camera is designed to be fixed over an eyepiece and clamped in position by means of pressure from its three adjustment screws, this pressure being applied to the outer casing of the eyepiece.

Although we have taken care to design this clamp mechanism to minimise the risk of damage to the body of the eyepiece (for example by providing a protective plastic sheath between the metal screw ends and the eyepiece) these measures will not eliminate all risk of damage and **some risk will always remain that pressure from the clamp mechanism may cause wear, scratches and/or indentation to the outer casing of the eyepiece.**

The user is advised to use minimum force when clamping the camera to an eyepiece and take whatever additional measures they see fit to reduce the risk of this external damage even further (e.g. by applying some protective coating between the eyepiece barrel and the camera clamp such as a thin layer of rubber sheeting or tape).

In our experience, using OptArc® microscope eyepieces, no visible damage is caused by careful use of this clamp mechanism but other makes and models of eyepieces may be have more fragile coatings or less rugged walls.

Do not use the clamp mechanism if the metal of the screws wears through the protective plastic sheath – damage will almost certainly occur to the eyepiece barrel in those circumstances.

By purchasing and / or using this camera the user agrees to accept responsibility for protecting their eyepieces from such damage. Dr Paul J. Tadrous, TadPath and OptArc cannot accept liability for any damage to eyepieces that may be caused due to the use or misuse of this product.

Safety certifications

The supplier of the electronic modules used in the AF51 states that the electronic components have FCC, RoSH and CE certification. We have not made independent tests on this so have not marked the camera with those certifications. The AF51 camera is supplied and sold on the understanding that this is an item of custom equipment for scientific research applications.

Contents

Download link for the PDF version of this manual

You can download the latest version of this User’s Manual as a PDF file from the Support section of the OptArc website via this link:

https://www.optarc.co.uk/support/

Video tutorials to accompany this manual

For video demonstrations on how to use this camera as well as the PARD Capture software see the PUMA Microscope YouTube channel:

https://youtube.com/@PUMAMicroscope

Customise the camera to fit your eyepiece

The CAD files for the 3D printed eyepiece clamp for this camera have been made open source so you can attempt to modify it and print your own version to fit an eyepiece that the standard clamp does not fit. See the Github page here:

https://github.com/TadPath/PUMA

Legal Information

Copyright

Trade Marks

OptArc and the OptArc logo are Registered Trade Marks of Dr Paul J. Tadrous (registered with the UK Intellectual Property Office).

Limitations of Use

This equipment is released to be used for research and educational purposes only. The AF51 camera is supplied and sold on the understanding that this is an item of custom equipment for scientific research applications.

Disclaimer

Only competent people trained in the use of handling of photographic, video, computer and optical equipment should use this product.

There is a risk of physical damage to the body / casing of eyepieces that have this camera clamped onto them. The camera must not be affixed to an eyepiece if such risk is unacceptable to the user or owner of the eyepiece. This notice is given here as fair warning. Dr Paul J. Tadrous, TadPath and OptArc cannot accept any liability for any such damage to eyepieces.

Dr Paul J. Tadrous, TadPath and OptArc cannot accept any liability for any loss or damages that may occur from the misuse or mis-handling of this product or from the failure to heed and abide by the warnings and safety notices set out in this document.

All information in this document is believed to be accurate according to the research and understanding of the author and is provided in good faith and in the hope that it will be useful. However, errors and omissions may be included herein. If any of this information is considered important to some application the user should verify that information from independent sources. Dr Paul J. Tadrous, TadPath and OptArc cannot accept any liability for any loss or damages that may occur due to errors or omissions contained in this document.

The descriptions and data presented in this manual only pertain to the genuine OptArc AF51 camera sold via OptArc.co.uk. Similar looking clones may claim to share many or all of the features of the AF51 but may not respond or perform in the way described in this manual.

Abbreviations

3D	3-Dimensional
AC	Alternating Current
ADC	Analogue-to-Digital Converter (or Conversion)
AE	Ali Express (an online marketplace)
AEC	Automatic Exposure Control
API	Application Programming Interface
AWB	Automatic White Balance
BH2 BHS	A brand of microscope from Olympus
bpp	bits per pixel
BMP	MS Windows BitMaP (image file format)
CAD	Computer-Aided Design
CCD	Charge-Coupled Device
CMOS	Ceramic Metal Oxide Semiconductor
DMLS	A brand of microscope from Leica
EFI	Extended Focal (plane) Imaging
FDM	Fused deposition modelling (method of 3D printing)
FITS	Flexible Image Transport System (file format)
FOV	Field Of View
FPS	Frames Per Second
gcc	GNU Compiler Collection (C compiler and linker software)
GNU GPL	GNU General Public License
HD	High Definition
HSI	Hue Saturation Intensity
JPEG	Joint Photographic Experts Group (compression standard for digital photos)
LENC	A pattern of variable gain across the image sensor designed to correct for uneven illumination of a lens (LENs Correction). This pattern is fixed in the sensor chip hardware (i.e. it does not change over time) and cannot be modified by the user.
MIT	Massachusetts Institute of Technology

MJPEG	Motion JPEG (compression standard for digital video)
MP	Mega Pixels (= millions of pixels)
MS	Microsoft
OS	Operating System
PARDUS	Programmable Affordable Remote-controlled Drive Utility Standard
PDF	Portable Document Format
PGM	Portable Grey Map (image file format)
PLA	Poly-Lactic Acid (3D printer filament plastic)
PLL	Phase Locked Loop
PNG	Portable Network Graphic (image file format)
PPM	Portable PixMap (image file format)
PUMA	Portable Upgradeable Modular Affordable (open source 3D printed microscope system)
RGB	Red, Green and Blue
SDK	Software Development Kit
SNR	Signal-to-Noise Ratio
SOC	System On-a Chip
TV	TeleVision
UNC	UNified Coarse (screw thread standard)
US	United States (of America)
USB	Universal Serial Bus
UVC	USB Video Class (a communications protocol standard for video devices)
v4l2	Video for Linux 2 (programming API for video devices in Linux)
VGA	Video Graphics Array resolution of 640 (wide) x 480 (high) pixels
WF	Wide Field
YUYV	A format for uncompressed digital video data

1. Introduction

What’s included in the box

The AF51 package sold by OptArc.co.uk includes the following components (figure 1.1):

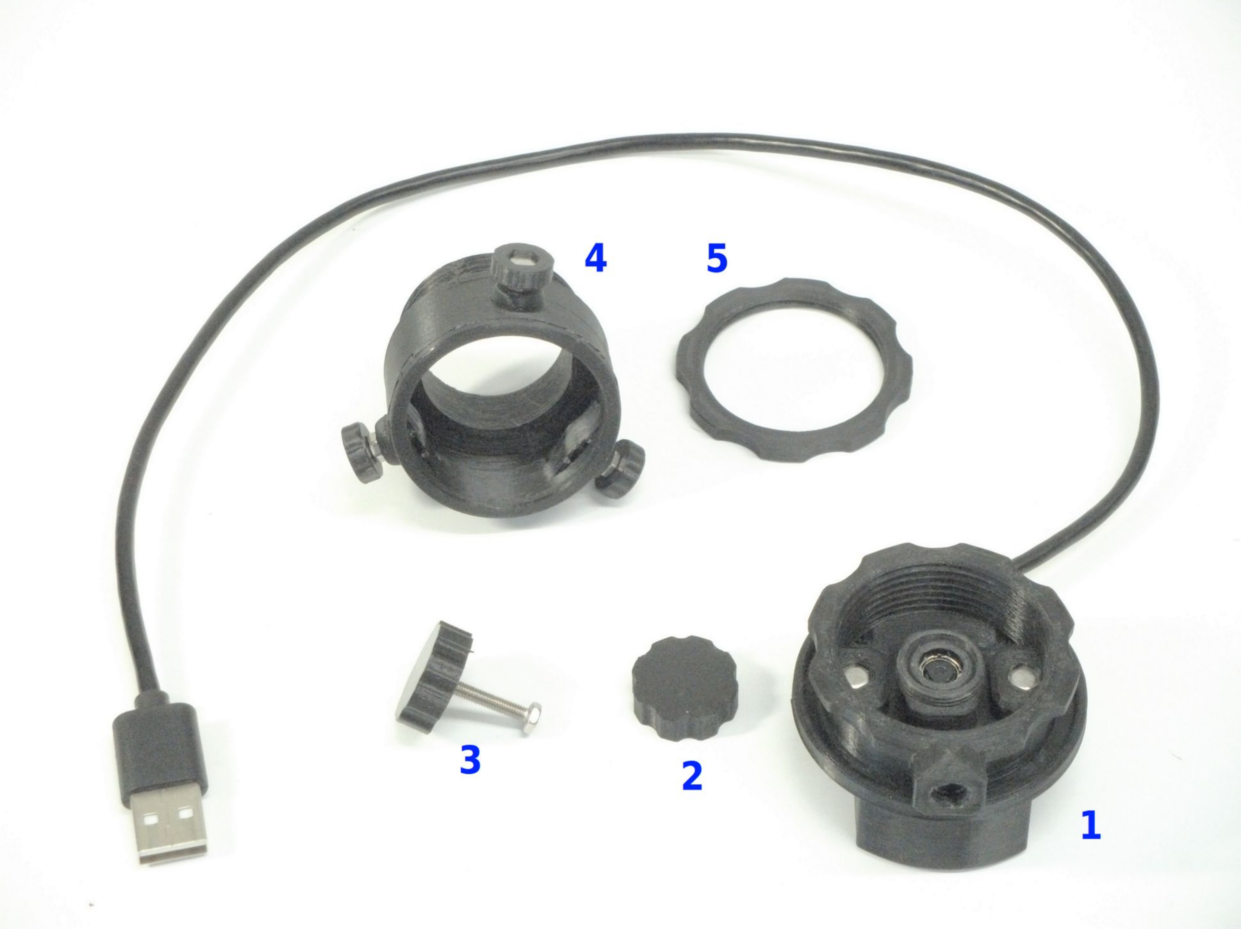


Figure 1.1 The components of the standard AF51 camera kit supplied by OptArc. The numbers refer to the items listed in the text. These are the only components supplied. All other accessories shown in this manual need to be purchased or otherwise acquired separately.

1. AF51 camera body with two centration adjustment screws and a female tripod thread socket to accept a standard 1/4-20 UNC threaded tripod screw (a tripod is not included) and a ‘Type A’ USB male connector on the end of an approximately 42 cm long USB cable.

2. A protective lens cap shown here removed from the lens. The camera will ship with this lens cap in place and the user must unscrew it to begin using the camera (see figure 1.2)

3. An adjustment screw tightening / loosening tool

4. Tri-screw adjustable eyepiece clamp.

5. Lock nut for fixing the position of the eyepiece clamp on the camera body.

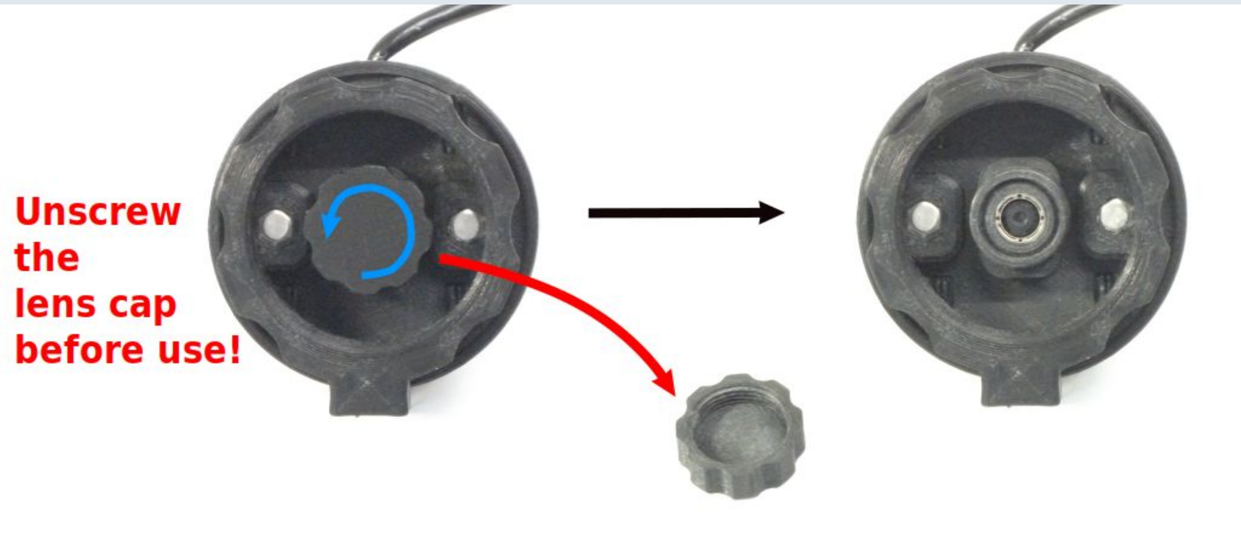


Figure 1.2 Unscrew and remove the lens cap before first use! The camera comes with this protective cap fitted. Re-fit the lens cap after use to protect the lens from dust. The lens cap can also be used to shield the lens from light when performing certain procedures such as collecting a masker dark frame for image processing. When putting the lens cap back on, take care not to over-tighten.

Overview of the AF51

The OptArc AF51 is a USB 2.0 CMOS colour camera specially designed to be suitable for use with the PUMA 3D printed open source microscope system (figure 1.3) but it may also be used with other optical equipment that use an eyepiece such as other microscopes and telescopes.

It has its own integral lens with user-programmable focus so may also be used for normal photography and video recordings, as an industrial inspection camera, for machine vision projects, as a document camera, a web cam, etc..

The camera requires a separate computer (not supplied) with a USB 2.0 compatible interface to operate and make recordings – it has no in-built recording capability or memory card slot. It is powered by the host computer via USB power. This camera does not have a microphone and cannot capture audio.

The camera functions and image capture may be controlled by your own programs using various SDK and API that are available for Linux, MS Windows and macOS.

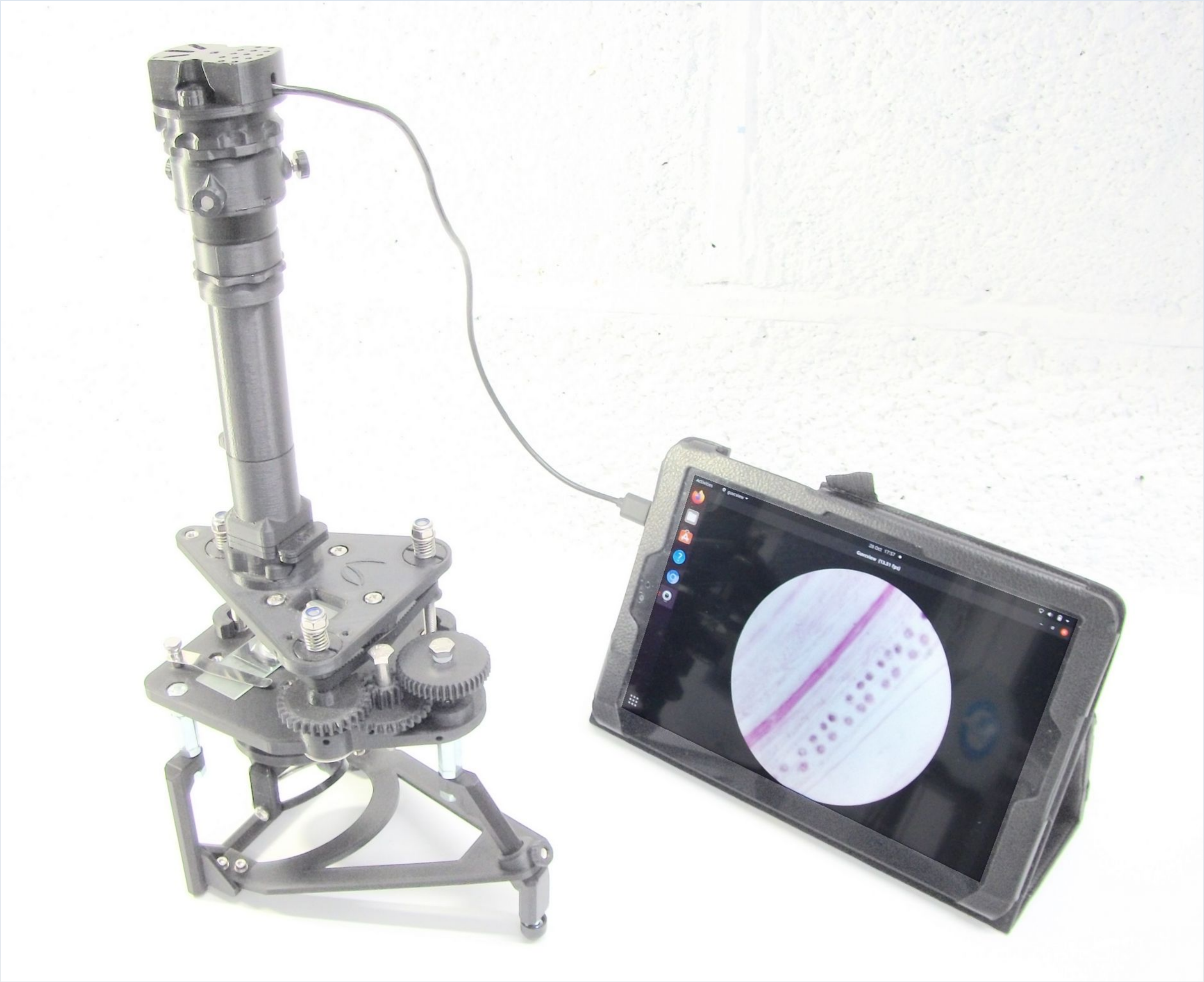


Figure 1.3 The AF51 in use on a PUMA microscope showing a live video image of the specimen under the stage by means of a tablet computer (microscope and computer are shown for illustration, they are not supplied with the camera). Note that the camera shows the complete circular field of view captured by the OptArc x10 WF high eye point eyepiece used in this setup.

Detailed specs will be given in a separate chapter but the main features that make the AF51 particularly suitable for use with the PUMA system is that it is small and light weight. This minimises the tendency to cause vibrations on the scope and fits in with the 'portability' ethos of the PUMA system.

The camera is designed with a built-in lens to image via afocal eyepiece projection and the lens system is designed to give a complete circle wide field of view when used with an OptArc® x10 WF high eye point eyepiece. The field of view with other eyepieces will vary (see chapter 5 for details of field of view with a range of different eyepieces).

Despite its small size, the AF51 boasts some high end features that make it a very versatile camera for both academic research, educational and recreational uses. Such features include its very wide dynamic range (including a 'night mode' which allows for multi-frame integration on the sensor chip) which make it useful for bright conditions as well as medium low light level applications such as fluorescence microscopy and some forms of astrophotography. It has high spatial resolution (up to 5 megapixels) so if you don't want a circular field of view in your photomicrographs you can crop this down to a filled rectangle while maintaining useful pixel resolution. It also has the ability to provide raw YUYV output and manual control over exposure and focus which are important for scientific image processing like Z-stacking extended focal imaging (EFI), deconvolution, quantitative microscopy and some forms of astrophotography.

Another bonus feature is that the casing is 3D printed and the design files for the eyepiece clamp system are available, free of charge, as open source FreeCAD files so a user with access to 3D printing (either DIY printing or an online 3D printing service) can customise the camera to fit a range of eyepieces and optical instruments that would not otherwise be able to fit to the standard eyepiece clamp provided with the AF51. More will be said about this in chapter 5 of this manual.

2. Recommended uses and limitations

No one camera is ideal for every application and so there are some important limitations of this camera. First here is an overview of what this camera is particularly good for.

Recommended uses of the AF51

Extra wide field of view afocal eyepiece projection

The AF51 is excellent for extra wide field of view imaging where you want to capture everything an observer sees with their eye against the eyepiece as illustrated in figure 1.3. This is in contrast to bare chip C-mount cameras which typically ‘crop’ the field of view to a highly magnified limited central rectangular region. This is the case for the OptArc x10 WF eyepieces and many other types but with some extra-wide field eyepieces the circular field will be cropped on top and bottom. These eyepieces therefore use more of the available pixels of the camera for their image of the specimen at the expense of not seeing the whole circular field of view. Details of how much of the camera sensor is used by the the eyepiece field of view with various eyepieces is given in chapter 5.

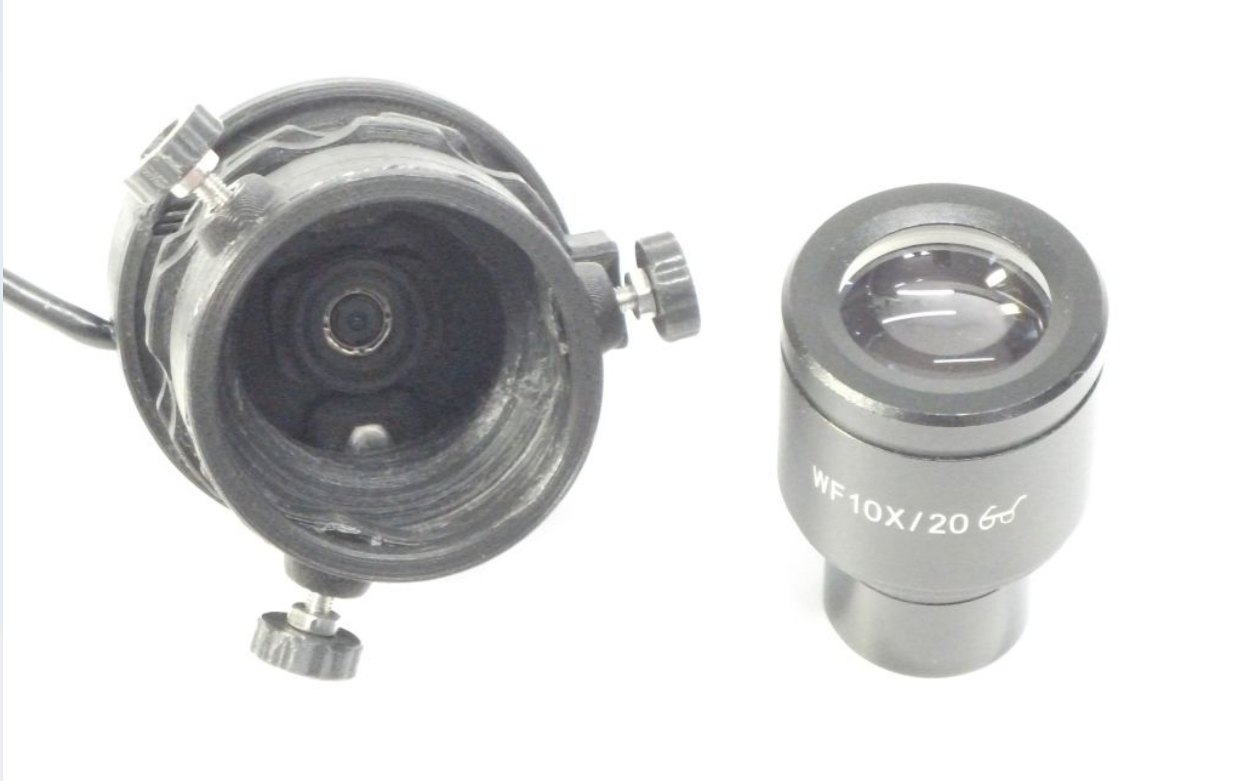


Figure 2.1 The AF51 is recommended to be used with high eye point wide field of view eyepieces such as the OptArc WF 10X/20 eyepiece shown in the figure. Its wide exit aperture and high eye relief allow the AF51 to capture the whole visible field of view with minimal wastage of peripheral pixels (see figure 1.3 for an example of this in use).

Colour imaging in bright conditions

This camera can give good colour imaging in bright conditions and at its maximum frame rates. The maximum frame rate achievable depends on the spatial resolution setting and exposure setting as well as whether raw YUYV camera output is used or whether MJPEG compressed camera output is used (both options are available with the AF51) and there are other limiting factors such as the speed of the host computer and the current state of the USB 2.0 connection. Detailed specs will be presented later but, by way of example, for raw YUYV output a maximal frame rate of 25 FPS is achievable at VGA resolution (640 x 480) and up to 3 FPS at full 5 megapixel resolution (2592 x 1944). If raw output is not required then selecting the MJPEG compressed image stream will allow all resolutions including full 5 megapixel frames to be streamed at a maximum rate of 30 FPS over a good clear USB 2.0 connection in a computer with sufficient free processing power and memory to handle this rate of data transfer.

The very best quality colour imaging with the AF51 – under any lighting conditions – is achieved using the monochrome signal from the camera (the Y signal) and taking 3 pictures, each with a separate primary colour filter in the light path. For more detail about this see chapter 7.

Medium low light level applications

The AF51 can also be used in medium low light level applications including dark field microscopy and immunofluorescence. Achievable frame rates will be lower due to the need for longer exposure settings and multi-frame integration (on the sensor chip) and averaging (in software) for the lowest end of the useable light level scales. At the lowest light settings the monochrome signal from the camera (the Y signal) will give the best results as opposed to using its in-built colour capability. This is partly because the colour signal processing inside the camera is subject to some artefacts that show up in low light conditions. See chapter 7 for more detail.

General and macro photo / video recording

The AF51 can also be used for ‘ordinary’ photography and video recording using it’s in-built lens i.e. not through a microscope or telescope (figure 2.2). Its lens has an auto-focus option that makes it useful for filming objects that alter their distance from the camera and it can also do macro photography of objects close to the lens.

For non-quantitative imaging the MJPEG compressed image stream of the camera can be used that can give up to 30 FPS at full HD (1920 x 1080) resolution as well as even higher resolutions such as 2K quad HD (2560 x 1440) and even up to the full 5 MP frame size (2592 x 1944). Bear in mind that **the camera has no microphone so if you need audio with your video you will need to arrange to record that with a separate**

microphone and add the audio to the video in post-production. The achievable frame rates and video resolution in practice will also depend on the processing power of the host computer you use to do the recording as well as the current speed of the USB 2.0 connection (e.g. whether the USB 2.0 hub is being shared with other devices, etc.).

Example uses include filming demonstrations of products or techniques, use as a document camera, use for machine vision applications (e.g. industrial inspection of products for quality control, facial recognition, robotics), use as a web camera, etc. whenever the wide field of view of this camera is advantageous to that task.

The camera can also be used for time-lapse sequences e.g. of natural phenomena or for filming 3D printing.

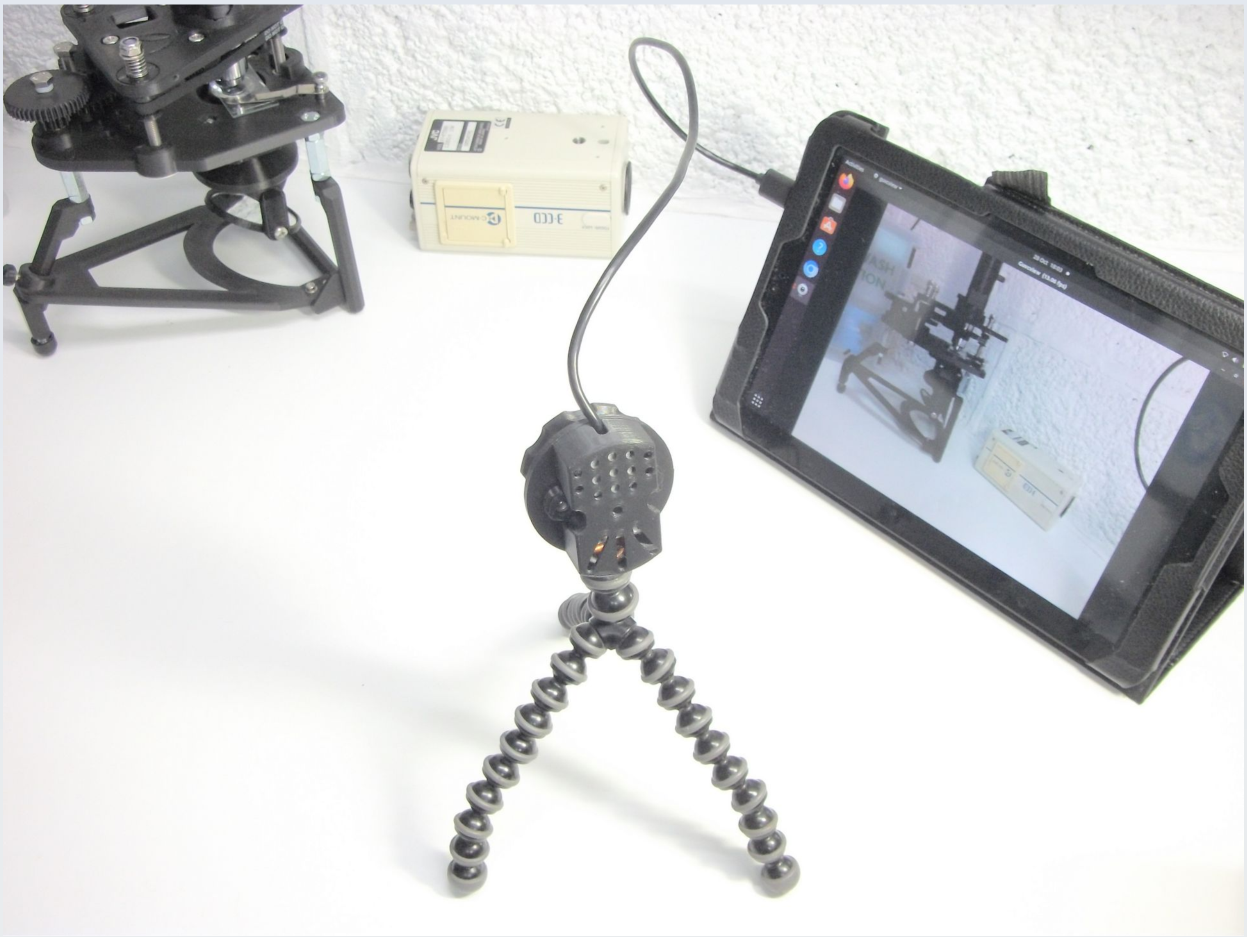


Figure 2.2 The AF51 being used as a 'standard' camera using its in-built lens (no eyepiece or other optics attached). The tripod and tablet computer are optional extras, not included with the AF51 package.

For macro photography (close-up imaging) you can focus up to 3 cm from the in-built lens without additional optics. This results in a field of view of about 39 mm x 29 mm using a 4:3 aspect ratio image format. Depth of focus will be quite narrow at this setting (there is no physical iris aperture to adjust field depth). Figure 2.3 shows an example image captured with this macro setting.

With macro photography you may find that adequate lighting becomes an issue because the object is so close to the camera body.

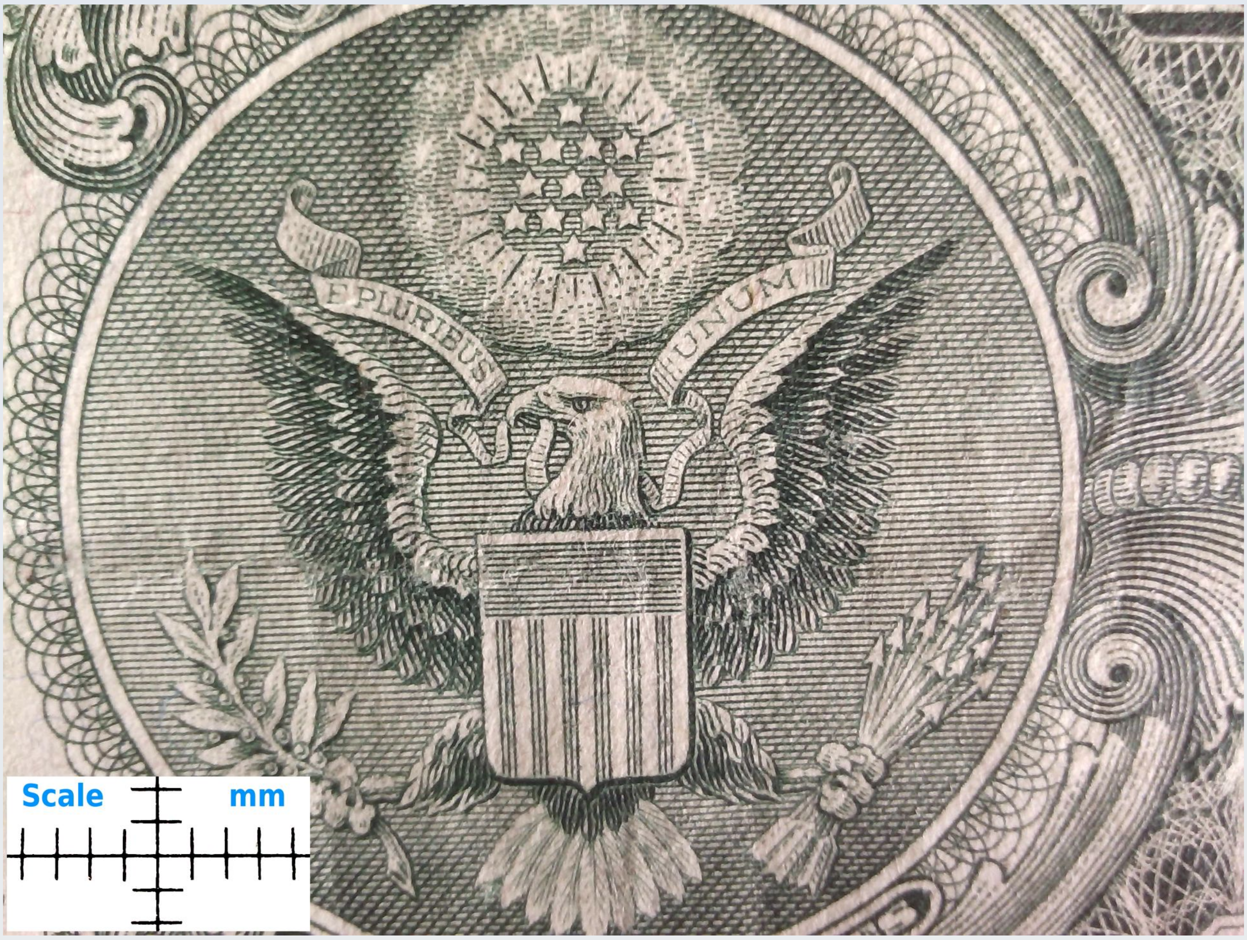


Figure 2.3 Part of a US 1 dollar bill captured with the AF51 in its macro mode. Note that images shown in this manual are compressed and down-sized for the PDF so do not represent the full quality of the original image taken from the camera. This figure just gives an idea of the scale of the macro setting.

Limitations of the AF51

Highest magnification imaging

The wide fixed angle lens and very wide field of view mean that this camera is not recommended where extreme high magnification is required. For example, it is not the best camera to use if you want to image subcellular organelles or study the detailed morphology of chromosomes or bacteria.

Short eye-relief or narrow field eyepieces

It is not recommended to use the AF51 with eyepieces that have a narrower angle field of view than the OptArc WF 10X/20 oculars. Attempting to use the AF51 with narrower field of view eyepieces will result in a very small field of view image disc with wastage of much of the sensor area thereby giving a relatively low resolution in the useable part of the image. Chapter 5 gives details of the amount of sensor area used by the field of view for a range of different eyepieces.

The eyepiece clamp that comes with the AF51 is designed to be used with eyepieces that have long eye relief. Some older or cheaper modern eyepieces with a small diameter exit lens and low eye relief will need a custom mount to be used with the AF51 (not supplied) because the standard eyepiece clamp that comes with the AF51 will not allow a close enough distance to be set between the camera and the eyepiece to see the full aperture. This results in a very small field of view and excessive vignetting. See chapter 5 for more details and measurements for various eyepieces.

Extreme low light level imaging

The AF51 is not recommended for extreme low light applications such as faint bioluminescence, deep sky faint object astrophotography or low light level imaging of fast moving objects.

This is because the AF51 uses its own lens with a small aperture and also because it images down an eyepiece using afocal projection photography. This means there will be a lot of lens elements between the light coming from the object and the CMOS sensor chip and each of these absorb, reflect or otherwise restrict some of the available light. Also the AF51 has a small CMOS sensor chip which limits its light gathering ability and does not have active cooling so dark current noise will be more prominent.

Although the AF51 can be used for some low light level applications, it will be working at maximum exposure times and gain and require multi-frame integration and multi-frame averaging to get the best quality images at these low light levels. Therefore it will not be able to image fast moving action without significant motion blur at low light levels. Colour sensitivity will also be reduced at low light levels.

Why not just use my smartphone camera instead?

Modern smart phone cameras have many times more megapixels than the AF51 and can give good quality photographs for illustration purposes – sometimes even better looking pictures than with the AF51.

The main limitations in using a smart phone is that these imaging systems are not designed for scientific or quantitative imaging and they are heavy and awkward to mount, especially on a 3D printed PUMA microscope where they will require some additional means of support to support their weight and avoid distorting the microscope frame and aggravating image 'wobble' (this is not such a problem with standard metal lab microscopes).

Regarding quantitative imaging for scientific applications you will need access to an uncompressed image stream from the camera as well as the ability to save that image in an uncompressed format to avoid artefacts (see chapter 7 of this manual), the ability to measure a dark current image and flat field image with multi-frame averaging manual control of exposure, colour balance, etc. and the ability to maintain a good signal even in low light levels (see chapters 6 and 7 of this manual). All this can be done with the AF51 and PARD Capture software. If you can do all too that with your smartphone camera and have it well mounted on the microscope then I see no advantage of using the AF51 over such a device for taking the odd picture. However, there are other advantages. The AF51 costs less than a high end smartphone and you can use multiple AF51 mounted on multiple eyepieces on the same microscope due to its light weight and small size. You can kit out a class or lab with multiple AF51's without having to have multiple smart phones. You can leave the AF51 on a scope in a lab or culture hood or on a robot to image over long periods where you would not want to leave a smartphone. The ease of coupling the AF51 to a standard computer for an image analysis workflow and automation, the use of the PARD Capture software to control the AF51 with its many conveniences for scientific imaging, multi-platform SDK for custom programming, etc.

3. Detailed hardware specifications

The AF51 has a rated power consumption of 500 mW (supplied via the USB cable from the host computer). Together with its USB cable and eyepiece clamp it weighs only 66 g.

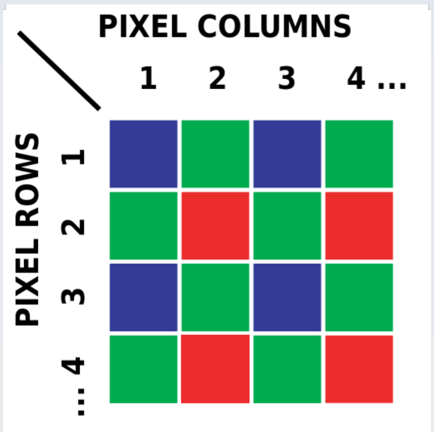
The camera is composed of three components:

1. The OmniVision® OV5647 CMOS sensor and lens system.

2. A microcontroller USB 2.0 interface board with custom firmware that allows the user to set various aspects of the sensor’s operation via the UVC protocol.

3. A 3D printed housing designed to allow practical use of the above two components.

The CMOS sensor and lens



	1	2	3	4 ...
1	🟦	🟩	🟦	🟩
2	🟩	🟥	🟩	🟥
3	🟦	🟩	🟦	🟩
... 4	🟩	🟥	🟩	🟥

Figure 3.1 The Bayer mask arrangement on the OV5647

- Sensor brand and type: OmniVision® OV5647 CMOS sensor with RGB Bayer filter
- Total pixels: 2624 columns x 1956 rows (5,132,544 total pixels) with a primary colour blue-green/green-red Bayer array arranged in line alternating fashion (see figure 3.1). The array includes active pixels (signals from which are read out to form the image) and inactive pixels (used for black level calibration).
- Active array size: 2592 x 1944 (5 MP)
- Image area: 3673.6 µm x 2738.4 µm
- Optical projection characteristics: Suitable for a lens system that produces a ¼ inch diagonal image projection area and a chief ray angle of 24°.
- Pixel size: 1.4 x 1.4 (µm)
- Well capacity: 4.3 ke^-
- Dark current: 16 mV/sec at 60 °C
- Temperature range (°C): Operational at -30 – 70, Stable image at 0 – 50.
- Fixed pattern noise: <1% of $V_{peak\text{-}to\text{-}peak}$
- Photosensitivity (without a lens): 680 mV/lux-sec
- SNR: 36 dB
- Dynamic range: 68 dB

- Shutter: electronic rolling shutter
- Maximum exposure interval: 1968 x t_{row} (where t_{row} = the minimum sampling period of the rolling shutter limited by the time-multiplexed ADC circuit)
- Proprietary OmniBSI™ (***B**ack **S**ide **I**llumination*) pixel architecture provides significant performance benefits over front-side illumination such as increased light sensitivity per unit area, improved quantum efficiency, reduced crosstalk and reduced photo-response non-uniformity.
- System-On-a-Chip (SOC) processing:
  - Field integration synchronous pixel readout with line-by-line transfer of the pixel data to an analogue amplifier to form a balanced signal ready for analogue-to-digital conversion (ADC).
  - 10 bit ADC provides 8 to 10 bit raw RGB data. The ADC can operate at up to 27 MHz and is fully synchronised to the pixel clock (see PLL below). The actual ADC conversion rate is determined by the frame rate.
  - Automatic exposure, white balance, band filter and black level control with manual overrides provided by firmware interface.
  - Programmable 16 zone size/position/weight control, cropping windowing and panning.
  - Lens correction ('LENC' i.e. a programmable fixed pattern of gain that varies across the sensor area to compensate for uneven illumination caused by the lens optics)
  - System clock control is via an on-chip phase locked loop (PLL) that generates a default 96 MHz clock from a 6-27 MHz input clock. An inside programmable clock divider is used to generate different frame rate timing. The full 5 megapixel frame is read at a rate 15 FPS with a pixel clock of 80 MHz. Lower resolutions are produced by cropping, subsampling or binning the full frames and are read at the following rates and pixel clocks: 1080p at 30 FPS / 68 MHz, 960p at 45 FPS / 91.2 MHz, 720p at 60 FPS / 92 MHz, VGA at 90 FPS / 46.5 MHz and QVGA at 120 FPS / 32.5 MHz. **Note that these FPS are the rates at which the CMOS SOC reads the frames and are NOT the maximum frame rates available from the AF51 camera** (those actual frame rates are dependent on the AF51 firmware and limitations of the USB 2.0 connection – see later for those frame rates).
  - Embedded 1.5 V regulator for core power

- Support for 2x2 pixel binning for resolutions lower than the maximum 5MP for better SNR in low light conditions
- Post binning resampling to minimise aliasing artefacts after 2x2 binning.
- On chip image windowing registers allow sub-windows of the full imaging area to be selected from the sensor (see the digital pan, tilt and zoom controls discussed below).

- Lens system
  - Effective Focal length: 2.8 mm
  - Field of view angle: 76 degrees
  - TV Distortion: <1%
  - F. number: 2.2
  - Relative illumination of sensor: 70%
  - IR Cut filter: 650 nm ± 10 nm (fixed)
  - Construction: 4 element
  - Focus: Electromagnetic voice coil mechanism allows for continuous auto-focus and programmable manual focus (figure 3.2).
  - Distance: 3 cm to infinity. The closest distances are most reliably achieved using manual focus.

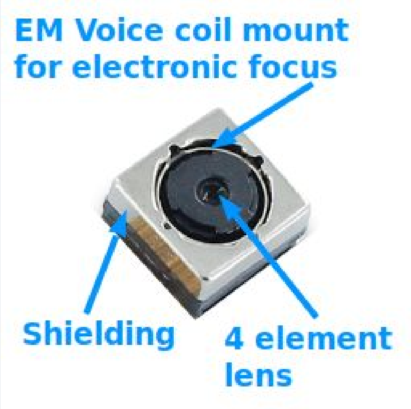


Fig 3.2 The lens system of the AF51 is contained in an 8.5 mm square metal package.

The USB 2.0 microcontroller interface

The interface board is a miniature PCB contained within the camera body (figure 3.3) and serves as the means by which images from the sensor chip are served to the software application on the user's host computer via the USB 2.0 UVC protocol. The camera has an approximately 42 cm long USB 2.0 cable that is connected to this microcontroller inside the camera body and the other (free) end of the USB cable has a 'type A' male USB connector for connecting to a host computer.

This microcontroller also serves as the means by which the user may communicate with the camera electronics to adjust certain parameters for imaging such as exposure, focus control, image resolution, image output format, etc..

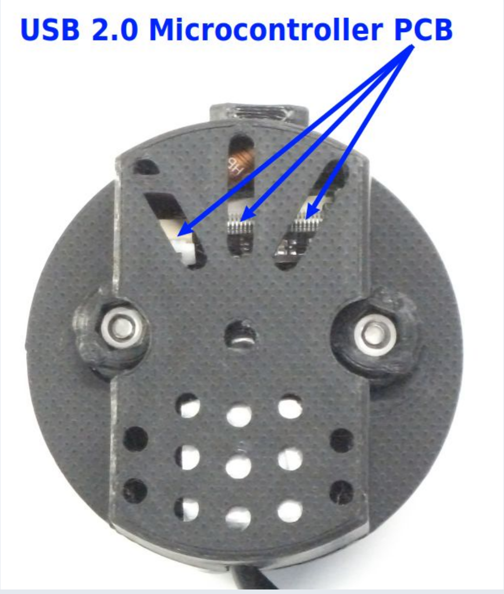


Figure 3.3 The USB 2.0 Microcontroller board in the AF51 camera housing.

While the sensor chip itself has many options for programming and adjusting it's SOC features, some of these options are only adjustable to hardware developers in the factory using specialist development interface boards. Not all of those features are programmable or controllable by the end user of the camera. For example the LENC gain pattern across the sensor has been programmed at the factory and cannot be altered by the end user via the the USB 2.0 interface board.

The features provided to the user by the USB 2.0 microcontroller interface are therefore the only parameters that may be adjusted by the user using suitable software on their computer (more will be said about the suitable / available software later). The following features are provided by the interface board for the user to adjust. There are some known 'quirks' or unintuitive behaviour of these settings which are explained in the notes in each section below.

The firmware in the interface board divides the controls into two groups: 'User Controls' and 'Camera Controls'.

User Controls

These control the image appearance by adjusting on chip signal processing. Table 3.1 shows the full set of available ‘User’ controls.

Control Name	Min	Max	Step	Default
Brightness	-64	64	1	0
Contrast	0	100	1	50
Saturation	0	100	1	64
Hue	-180	180	1	0
White Balance, Automatic	0	1	1	1
Gamma	100	500	1	300
Power Line Frequency	0	2	1	1
White Balance, Temperature	2800	6500	10	4600
Sharpness	0	100	1	50
Backlight Compensation	0	2	1	0

Table 3.1 ‘User Controls’ for the AF51 Camera. The minimum possible value, maximum possible value and default setting value are shown as well as the minimum step change that is permitted.

Notes:

The ‘**White Balance, Automatic**’ turns automatic white balance (AWB) on (1) or off (0). When on, the current value for ‘White Balance, Temperature’ will be ignored and cannot be set. When AWB is turned off, the value currently held in the ‘White Balance, Temperature’ will be applied. Varying that value will vary the white balance setting of the image.

The value for ‘**Power Line Frequency**’ corresponds to 3 menu options: ‘Disabled’ (0), ‘50 Hz’ (1) and ‘60 Hz’ (2). This works to reduce or eliminate horizontal sinusoidal intensity fluctuation lines that are sometimes seen with images taken with fast exposure times using AC filament bulb illumination in older microscopes (figure 3.4). **This will not completely eliminate AC-type light fluctuations in all circumstances** (for example it cannot eliminate flickering seen with some fluorescent tube sources – even when that flickering is regular).

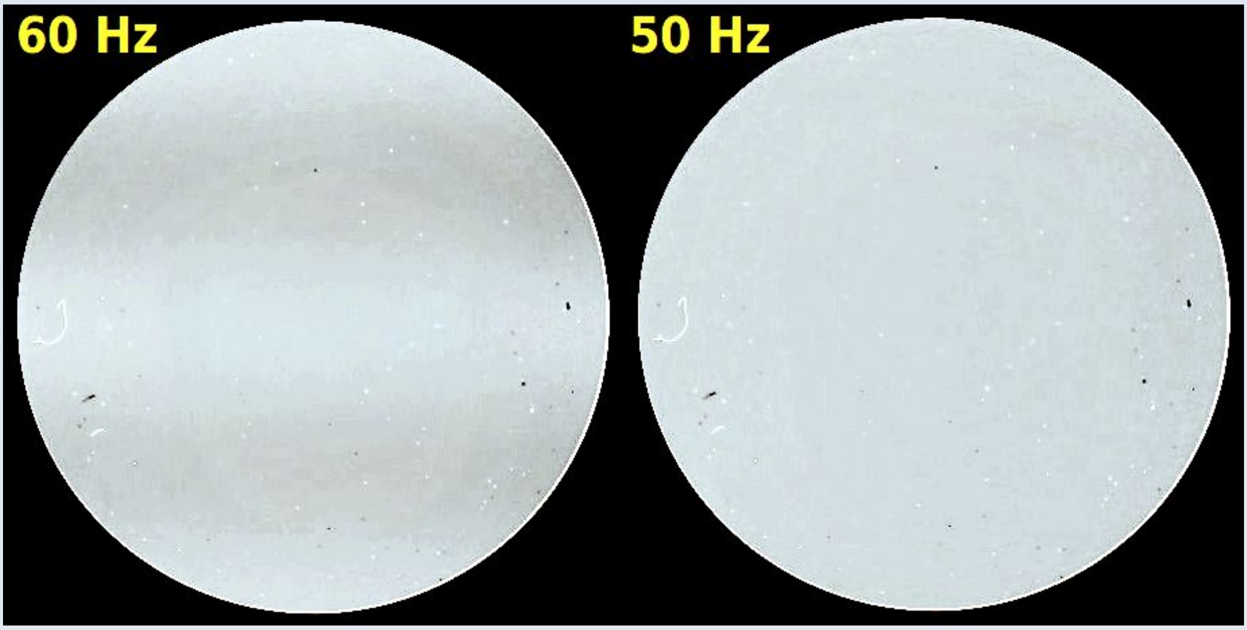


Figure 3.4 Example of the effect of the ‘Power Line Frequency’ settings (shown in yellow text at the top left of each image). The photographs were taken with the AF51 on a Leica DMLS microscope equipped with its original AC filament bulb operating on a 50Hz AC mains supply. The image is contrast enhanced to exaggerate the effect for the purposes of illustration.

The reduction in horizontal line fluctuations is best seen in live images. Still images taken with any of these settings may still show residual sinusoidal fluctuation. **Residual fluctuations can be reduced or eliminated by using multi-frame averaging (see software options later in this manual) or a longer exposure setting in the camera (see below).**

Note that it is ineffective to change this setting from value 2 (60 Hz) directly to value 0 (off). You need to change from 2 to 1 to 0. Going the other way does not matter (you can change from 0 directly to 3). The reason for this behaviour is that the ‘off’ (0) setting for this control simply leaves the setting where it was last – it is effectively non-functional.

The ‘**Backlight Compensation**’ feature affects the behaviour of the automatic exposure level when the automatic exposure control (AEC) is enabled (automatic exposure settings are discussed below in the section on the ‘Camera Controls’).

The AEC algorithm samples pixel values from target areas across the chip and attempts to increase or decrease the exposure of the chip to avoid saturation of pixels while using the maximum dynamic range of pixel values for the image. If the image is too brightly lit tending to cause saturation it will reduce exposure till there are no saturated pixels in its target reference areas. Conversely if the pixels in the target areas are too dark it will increase exposure. In this way the AEC attempts to use a maximal amount of the pixel value range for the image without over or under-exposing the scene.

In some scenes there may be too much light in peripheral zones that would tend to make the AEC under-expose a subject in the centre of the field of view and *vice versa*. The 'Backlight Compensation' setting tells the AEC algorithm to ignore the values of a certain amount of pixels in the peripheral areas of the chip when doing its calculations (the higher the setting, the more pixels are ignored). Thus a high 'Backlight Compensation' value results in relative over-exposure of the central area of the image and a zero value will reduce exposure of the central pixels by the AEC to what it would normally do without compensation (see figure 3.5).

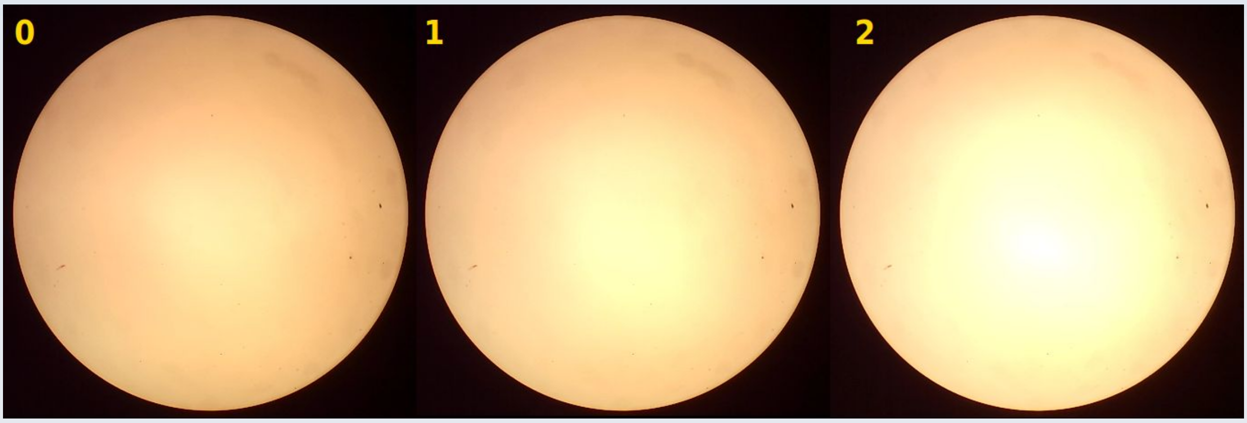


Figure 3.5 Backlight Compensation factors of 0, 1 and 2 (from left to right) on an unevenly lit background from a microscope using the AF51 in AEC mode (i.e. 'Aperture Priority' exposure mode).

From the above description it should be clear that changing the 'Backlight Compensation' setting when the camera is NOT in an automatic exposure mode will have no effect. Also the effect of the 'Backlight Compensation' setting on the AEC level will vary depending on the image currently falling on the sensor chip.

You can adjust '**brightness**' and '**contrast**' with the controls of those names but these only change the post capture amplifier properties and do not increase light gathering abilities of the chip.

Camera Controls

These adjust image acquisition parameters such as exposure, pixel binning, cropping and lens control. Table 3.2 shows the full set of available ‘Camera’ controls.

Control Name	Min	Max	Step	Default
Auto Exposure	0	3	1	3
Exposure Time, Absolute	50	10000	1	166
Exposure, Dynamic Framerate	0	1	1	1
Pan, Absolute	-57600	57600	3600	0
Tilt, Absolute	-43200	43200	3600	0
Zoom, Absolute	0	3	1	0
Focus, Automatic Continuous	0	1	1	1
Focus, Absolute	0	1022	1	68

Table 3.2 ‘Camera Controls’ for the AF51 Camera. The minimum possible value, maximum possible value and default setting value are shown as well as the minimum step change that is permitted.

Notes:

The ‘**Auto Exposure**’ can be set to 2 values: 1 (corresponding to ‘**Manual Mode**’) and 3 (corresponding to ‘**Aperture Priority Mode**’ which is the AEC). Although the camera firmware reports that a range of 0-3 is possible the values 0 and 2 cannot be set in the AF51 (they are redundant for this model).

The AF51 does not have a separate ‘gain’ control. The exposure setting (whether manual or automatic) actually sets both the gain and exposure simultaneously. The relative contribution of gain and exposure is calculated by means of an on-chip algorithm that is designed to maximise the SNR for any given lighting condition. See below for details.

Manual mode provides a constant, fixed, gain value and exposure time. This is set via the ‘**Exposure Time, Absolute**’ control value. This value can only be set or changed while the camera is in manual exposure mode (i.e. ‘Auto exposure’ is set to 1). Practical experience shows that the gain component in manual mode tends to be lower (so there is less noise) and the exposure component higher (slower frame rate) compared to an equivalent low light level sensitivity achieved with the AEC. However the AEC can achieve greater sensitivity overall than any manual setting (see below).

The ‘**Aperture Priority Mode**’ corresponds to automatic exposure (the characteristics of which can be modified via the ‘Backlight Compensation’ setting as described in the previous section) .

The automatic exposure control (AEC) algorithm adjusts both the exposure time and gain as part of its algorithm in an attempt to achieve an ‘optimal’ SNR. The AEC algorithm favours increasing the exposure time over increasing the gain when the image needs to be made lighter in low illumination conditions and it favours decreasing the gain over reducing exposure time when the image needs to be made darker. The gain and exposure settings of the AEC can be modified by use of the ‘Backlight Compensation’ setting (described above) and the ‘Exposure, Dynamic Framerate’ switch (described below).

The AEC algorithm is based on sampling the average luminance in 16 rectangular subregions within the whole imaging area (these subregions are arranged in a grid 4 regions across and 4 down).

In extremely bright conditions the AEC algorithm will enter into ‘***Light AEC***’ mode (LAEC) which allows integration time to be less than 1 row.

In extremely dark conditions the AEC will activate '***night mode***' which allows integration time to be larger than one frame. In night mode, the chip can integrate up to 8 frames by slowing down the original frame rate and waiting for exposure. This is what enables the AF51 to be used in medium low light level applications that would otherwise not be practical for a CMOS cameras of this type.

Night mode and LAEC cannot be set via the manual exposure control (they are beyond the limits of the ‘Exposure Time, Absolute’ range) but they can be entered into and fixed as a constant setting if the AEC is de-activated when either of these modes is currently operating. In this way they can be used as ‘one-touch’ fixed settings.

The ‘**Exposure, Dynamic Framerate**’ control is an on (1) / off (0) setting which only makes a difference when in auto-exposure (‘Aperture priority’) mode. When on, this enables the AEC to override the camera stream frame rate (selectable by the user – see later in this manual for the available frame rates) so as to reduce the frame rate even further for darker illumination scenes to allow longer exposure times. This effectively allows you to get the maximum light sensitivity in low light conditions when using certain camera format steams that have a faster frame rate setting. If you set this to 0 then the AEC range will be limited by the frame rate of the current camera image format stream – you will be able to image faster moving action but with lower light sensitivity. The effect of this setting remains constant if you switch off the AEC mode (by setting ‘Auto Exposure’ to 0) while ‘Exposure, Dynamic Framerate’ is active. In these circumstances, changing the setting ‘Exposure, Dynamic Framerate’ will have no effect until you set a value for ‘Exposure Time, Absolute’ or until you re-enter AEC mode by setting ‘Auto Exposure’ to 3.

The ‘**Pan, Absolute**’, ‘**Tilt, Absolute**’ and ‘**Zoom, Absolute**’ controls allow you to select a sub-window from the main image sensor area for display but only when the camera is in certain resolution modes. The conditions under which these controls are operational are shown in table 3.3. An example of the effect of varying the ‘Tilt, Absolute’ control value is shown in figure 3.6. This ‘Tilt, Absolute’ feature may be particularly useful when using the wide-format image resolutions in cases where the whole image sensor is used (so not with the circular field of view seen down an eyepiece) in order to select what part of the full 4:3 sensor image gets included in the captured 16:9 format image.

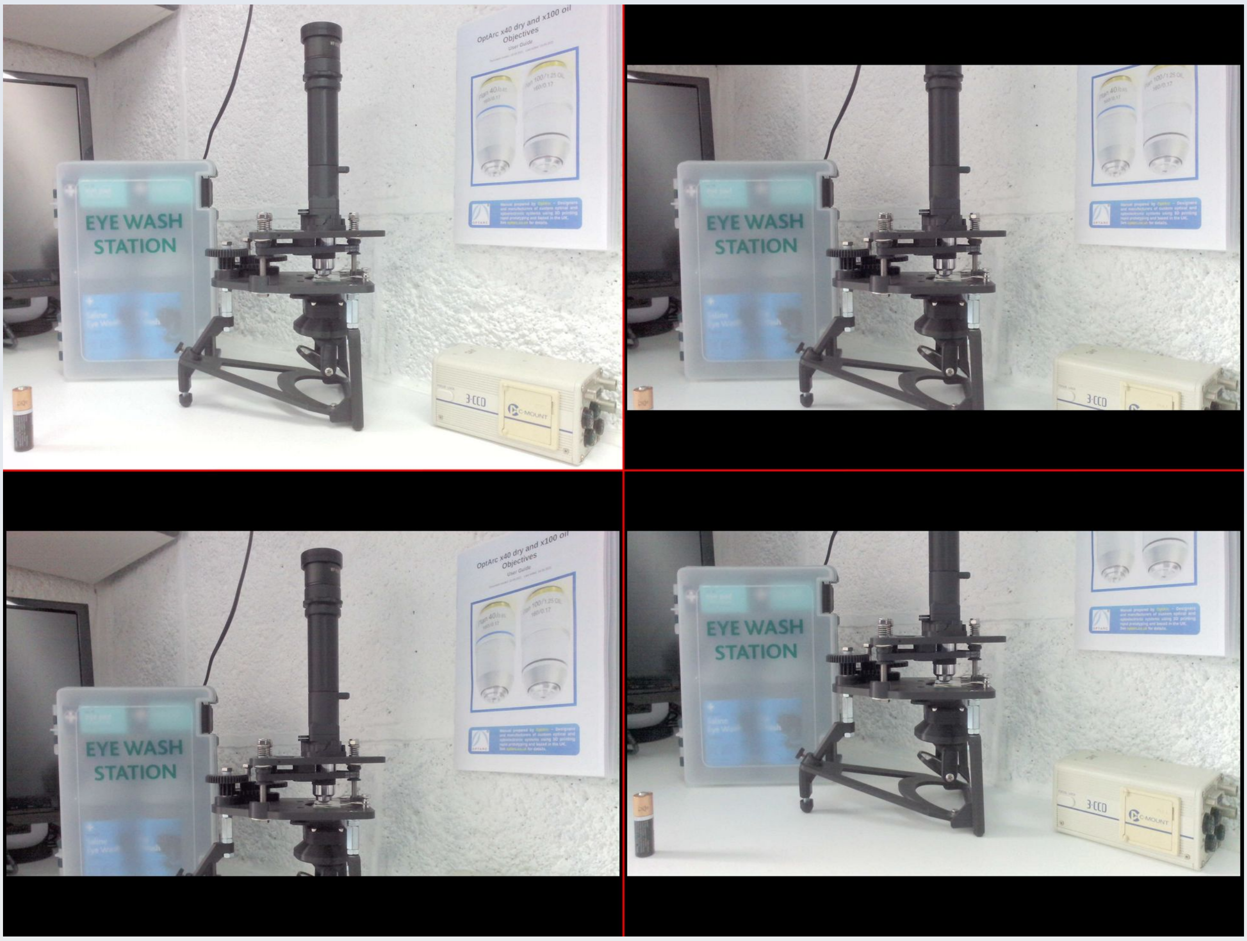


Figure 3.6 Effect of varying the ‘Tilt, Absolute’ control when in a wide format mode showing how this can select what part of the full sensor image to display in the wide format output. Top left is an image taken with a 4:3 aspect ratio image format which captures the full sensor data. Wide screen formats have to leave out some of the vertical sensor information. The default ‘Tilt, Absolute’ value of 0 (top right image) selects the central part of the full sensor image leaving out an equal amount of top and bottom information. Bottom left image is taken with a ‘Tilt, Absolute’ value of -43200 (maximum negative value) and captures the top part of the full frame whereas the bottom right image shows the result with a ‘Tilt, Absolute’ value of 43200 and this captures the bottom-most part of the full frame.

The ‘Pan. Absolute’ ability, where enabled only pans by a few pixels left or right for resolutions that are greater than VGA but panning by a larger amount is available when the ‘Zoom, Absolute’ feature is used. The ‘Zoom, Absolute’ function is only enabled when both height and width are smaller than the corresponding VGA values and even then the number of zoom steps is limited to give a maximum zoom that is equivalent to VGA resolution. There are a maximum of 4 zoom settings (0,1,2,3) that correspond to the following zoom magnification factors: 0=x1, 1=x1.28, 2=1.6, 3=x2.0.

Although this zoom function is ‘digital’ in that is does not alter the actual lens position or characteristics, it is not simply a mathematical upscaling of a low resolution image (as is often the case with ‘digital zoom’ in consumer cameras). Rather it ***does*** give a genuine increase in pixel resolution compared to the un-zoomed image. This is because the AF51 zoom mechanism works by sampling from the full 640x480 VGA output of the chip even when the chosen video stream size is lower resolution than this. The higher the zoom factor, the closer to 1:1 sampling of the full VGA image is produced (recall that zooming only works for image formats lower than full VGA). However, capturing such a zoomed image does not give quite as good an image quality as capturing in native VGA resolution without zooming (see figure 3.7).

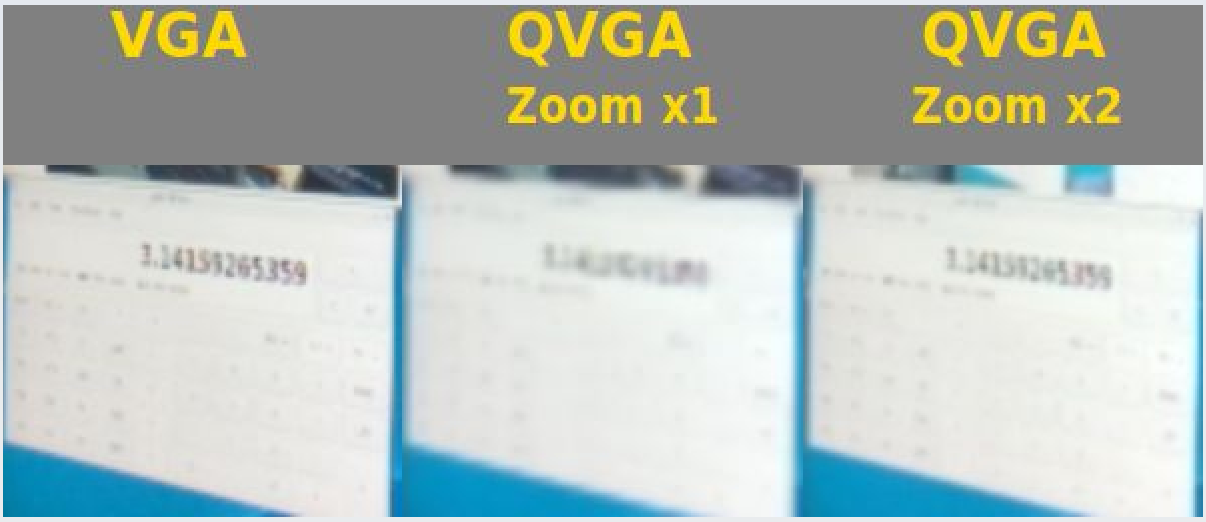


Figure 3.7 The effect of the ‘Zoom, Absolute’ factor on image quality for images that are below full VGA (640x480) resolution. The pictures are of part of a computer screen photographed with the AF51 which is at an angle to the screen – not face on). These show only a small part of a frame to emphasise the quality and resolution differences and each picture has been scaled to the same size as the others for comparison in this figure. While zooming does improve the pixel resolution of the QVGA (320x240) image, it is still not quite as clear as the capture from the native 640x480 image stream.

‘**Focus, Automatic Continuous**’ is an on (1) / off (0) control that sets whether the optical auto-focus feature of the camera lens is active or not. When active the lens will move in or out to main focus of a subject near the centre of the field of view. This auto-focus algorithm is not perfect so focus can sometimes shift in an undesirable way. When this setting is on the value for the ‘Focus, Absolute’ setting will be disregarded. When ‘Focus, Automatic Continuous’ is set to 0 (off) the lens will stay in the position it was last in during auto-focus until a value for ‘Focus, Absolute’ is sent to the camera. This means that ‘Focus, Automatic Continuous’ can be used as a ‘one-touch’ automatic focus control (i.e. you enable ‘Focus, Automatic Continuous’ and wait till the desired focus is achieved for you then you set ‘Focus, Automatic Continuous’ to 0 and that focus will be maintained and fixed until you either re-enable ‘Focus, Automatic Continuous’ or you send a value to the ‘Focus, Absolute’ control). In most circumstances ‘Focus, Automatic Continuous’ should be disabled (set to 0) when using the AF51 with an optical instrument coupled to an eyepiece and for proper afocal eyepiece projection coupling this fixed focus should be at infinity.

‘**Focus, Absolute**’ gives you manual (or programmatical) control over the focus to produce a fixed focus of the lens at a position determined by the value supplied for this control. The value in this control will only be settable or alterable when ‘Focus, Automatic Continuous’ is set to 0. The value for ‘Focus, Absolute’ can be varied from 0 (representing focus at or just beyond infinity) to 1022 (representing focus at about 3 cm from the front of the lens, the closest focus possible with this lens). For most work using the AF51 coupled to an eyepiece the focus should be fixed at infinity (at or near a value of 0). In practice I find that this voice-coil focus mechanism can get a bit ‘sticky’ in that if focus has been held fixed for some time then you might need to increase or decrease the focus value quite a bit before it actually changes the voice coil position (with a jolt) and, at that point, more fine continuous control is again restored.

Camera stream formats

The AF51 camera will stream images down the USB cable in a format selectable by the user via interaction with the USB 2.0 microcontroller interface. The user may select to get either raw images in the YUYV format (this results in slower frame rates ranging from a maximum of 3 to 25 FPS depending on the frame resolution but higher quality images best suited for scientific applications) or the compressed MJPEG format (this yields images at a maximum of 30 FPS at all resolutions so is best suited to capturing or previewing live HD video but with each image having a degree of image compression artefact).

The complete list of available formats and resolutions are shown in table 3.3.

Resolutions (WxH)	Aspect Ratio	Max. FPS YUYV	Max. FPS MJPEG	Also Called	Pan	Tilt	Zoom
320 x 240	4:3	25	30	QVGA	Yes (Z)	Yes (Z)	Yes (3)
424 x 240	16:9*	25	30	WQVGA	Yes	Yes	Yes (2)
640 x 360	16:9	25	30	nHD	Yes	No	No
640 x 480	4:3	25	30	VGA	No	No	No
848 x 480	16:9*	15	30	WVGA	Yes	Yes	No
960 x 540	16:9	10	30	qHD	Yes	Yes	No
1280 x 720	16:9	10	30	HD 720p	Yes	No	No
1280 x 960	4:3	10	30	SXGA-	No	No	No
1920 x 1080	16:9	5	30	Full HD 1080p	No	No	No
2560 x 1440	16:9	3	30	QHD 2K 1440p	Yes	Yes	No
2592 x 1944	4:3	3	30	5 MP	No	No	No

Table 3.3 Available camera stream formats for the AF51. Also shown is the compatibility of each format with the (digital) Pan, Tilt and Zoom controls of the camera. * These are the closest approximations to 16:9 aspect ratio that maintain modulo 16 compatibility (i.e. each dimension pixel count is exactly divisible by 16). This is important because encoding frames into a streaming format is done by breaking the frames down into blocks of 4x4 or 16x16 pixels. In the above names Q can stand for ‘quarter’ (meaning 4 times fewer pixels) or ‘quad’ (meaning 4 times more pixels) and ‘q’ stands for ‘quarter’ (qHD has a quarter the number of pixels as 1080p Full HD); W indicates the wide version of a format. nHD stands for ninth HD because it has 9 times fewer pixels than 1080p Full HD. VGA refers to the Video Graphics Array standard of 640x480 pixels, SXGA- refers to an unofficial terminology to indicate a slightly lower resolution version of the Super eXtended Graphics Array standard (the original SXGA is 1280x1024). Sometimes QHD is also referred to as ‘2.5 K’. The QHD (quad HD) terminology refers to the fact that it has 4 times the number of pixels as ‘standard’ HD (720p) resolution. The Pan, tilt and Zoom entries show which of these functions are available at each resolution. ‘Yes (3)’ means all 3 steps of zoom can be done, ‘Yes (2)’ means only two steps of zoom can be done. ‘Yes (Z)’ means pan or tilt will work only when zoomed in.

Notes:

In general, the higher the spatial resolution and the lower the frame rate, the greater is the maximum possible light sensitivity of the camera. This is because with higher resolution comes more rows of pixels and it takes longer for the rolling shutter to read through all those rows of pixels before getting back to the first row meaning exposure interval for all rows is longer. Also, the lower the frame rate, the longer interval between frames which allows each frame to expose for longer.

This has the consequence that for the best low light level imaging, the camera cannot record fast moving objects clearly (they will appear as a streak across the field of view if detected).

The '**Max. FPS**' shown in table 3.3 are the maximum possible frame rates achievable under ideal conditions of adequate illumination, fastest exposure settings and a good clear USB 2.0 connection to a computer with sufficient processing speed, read-write speed and memory capacity. The actually achieved frame rate may be slower due to limitations in any of the above factors such as a shared or busy USB 2.0 hub, slow, delayed or queued processing by the host computer, low light levels resulting in longer frame exposure times, etc.

3D Printed housing

The sensor chip and interface board are set in a 3D printed enclosure made with PLA filament and designed to enable adjustable mounting of the camera on a variety of eyepieces or on a standard tripod and to provide heat sink passive cooling for the sensor chip and interface board. The casing is made with an Ender 3 FDM 3D printer.

Heat sink enclosure

The camera chip can get very hot especially when streaming large format images. If this heat is not effectively dissipated it can cause malfunction of the camera. Also, the hotter the sensor chip, the more dark current noise is present in the image signal which adversely affects the ability to capture high quality images at low light levels.

For all these reasons the camera housing is specially designed with carefully placed vent holes as well as an internal aluminium and steel passive double heat sink system what is directly applied to the back of the sensor chip module (see figure 3.8). Part of the heat sink foil is directly applied to the back of the casing over some of the vent holes to radiate heat directly up through these holes and be cooled by air currents entering from the other holes. The free holes allow circulation of air within the housing to cool both the sensor chip and the interface board components.

The central hole goes right down to an area of the heat sink foil that is in direct contact with the metal plate on the back of the sensor chip module and this is held there with an internal spring and makes contact with it via a layer of thermal transfer compound.

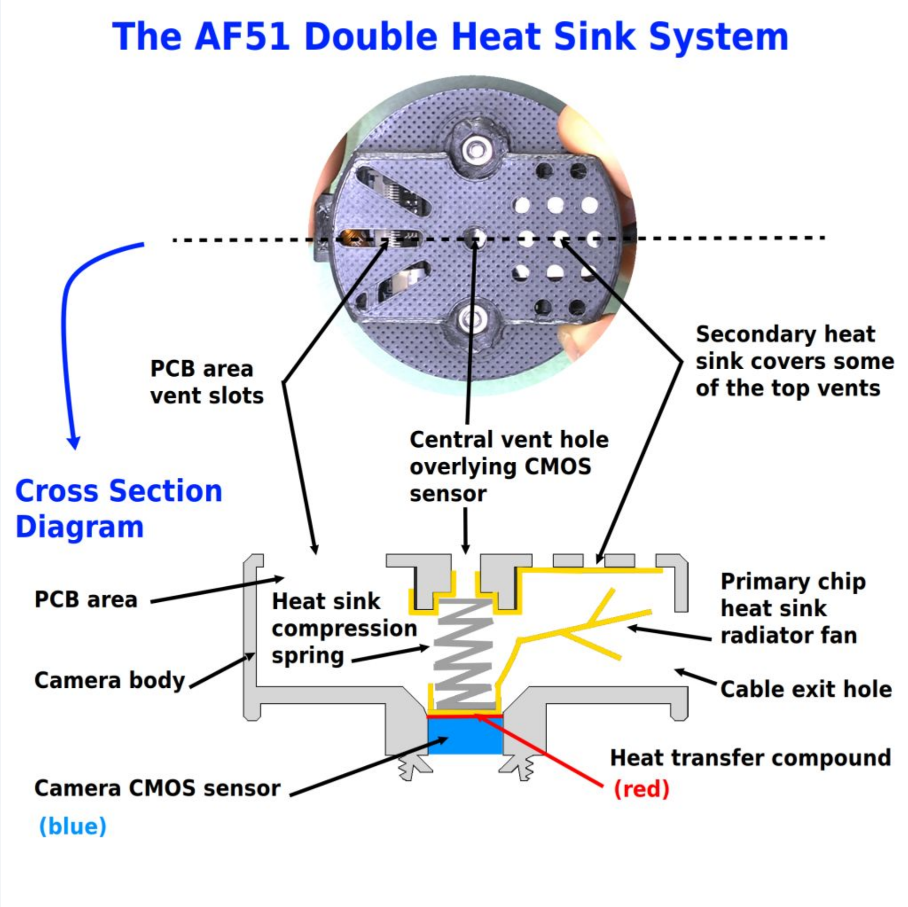

The AF51 Double Heat Sink System
PCB area vent slots
Secondary heat sink covers some of the top vents
Central vent hole overlying CMOS sensor
Cross Section Diagram
PCB area
Heat sink compression spring
Primary chip heat sink radiator fan
Camera body
Cable exit hole
Heat transfer compound (red)
Camera CMOS sensor (blue)

Figure 3.8 The AF51 passive double heat sink system. The steel spring has a dual action - to keep the aluminium radiator firmly applied to the hot back plate of the CMOS sensor and to act as a heat sink in its own right, conducting heat up to the second aluminium foil radiator under the central vent holes. These components are held in place by the pressure of the spring so the casing must not be opened or they will come apart. Yellow = aluminium heat sink foil.

There are no user-serviceable parts or adjustment controls inside this enclosure. **Under no circumstances should this casing be opened otherwise the spring holding the heat sink system against the back of the sensor chip will come loose and the heat sink will loose its effectiveness resulting in malfunction of the camera**. For this reason also **avoid handling the camera roughly or pulling at the back of the camera** (e.g. when trying to remove the camera from an eyepiece) – **always hold the camera system from the eyepiece clamp or the translation adjustable mount** (both to be described below).

Likewise, **under no circumstance should any of the vent holes be blocked and do not poke any instruments in through the holes or this may damage the heat sink foil (which is very thin in places) and/or risk short circuiting the camera**

electronics. Also due to all these vent holes **the camera must not be operated in the rain or other wet conditions – it is not water resistant or splash proof**.

Translation adjustable mount

The camera heat sink enclosure is connected to a mount that has a thread to accept the eyepiece holding clamp (this clamp will be described later). This mount is held against the flat part of the camera heat sink enclosure by two M3 machine screws that rise up through over-sized holes which allow a degree of XY shift freedom in this fit (see figure 3.9). The inscribed circles printed onto the flat part of the camera heat sink enclosure serve as a guide to help you know how nominally centred is the translation adjustable mount relative to the camera heat sink enclosure.

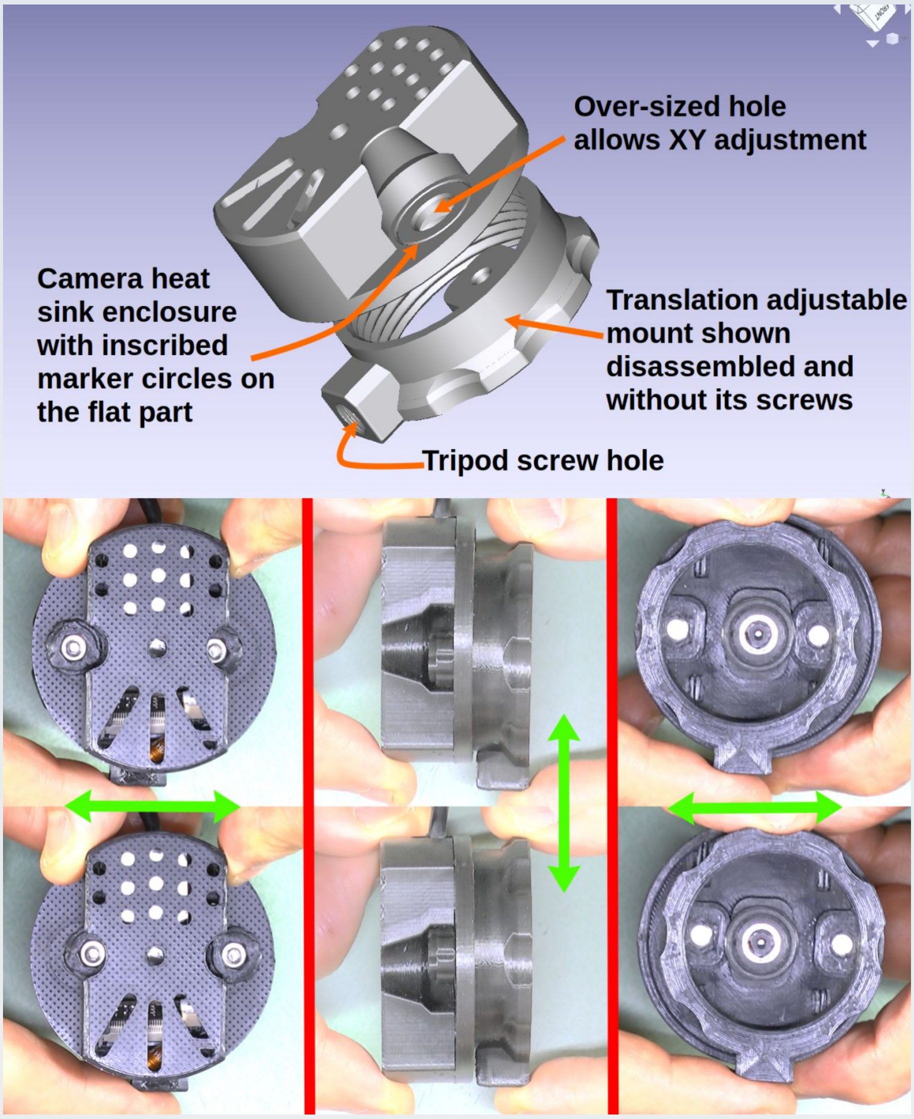

Over-sized hole allows XY adjustment
Camera heat sink enclosure with inscribed marker circles on the flat part
Translation adjustable mount shown disassembled and without its screws
Tripod screw hole

Figure 3.9 The translation adjustable mount allows the user to centre the camera over any eyepiece. Top panel shows the mount disassembled from the camera heat sink enclosure with the M3 screws and thumbscrew nuts not shown. The lower panel shows the assembled parts with examples of the possible XY translation motion from the top, side and bottom.

This arrangement provides a mechanism to adjust centration of the camera lens over an eyepiece when one is fitted. When the desired centration position is found the mount can be fixed in this position by tightening the two ‘thumbscrew’ nuts against the flat part of the camera heat sink enclosure using the adjustment screw tightening / loosening tool (item 3 in figure 1.1). This is done by inserting the metal hexagonal part of the tool into the hexagonal depression in the thumbscrew nut and rotating the tool with your fingers to tighten or loosen the thumbscrew nut (figure 3.10). **Do not over-tighten or you may damage the tool** – only tighten enough to prevent accidental movement of the mount off the desired XY centration position.

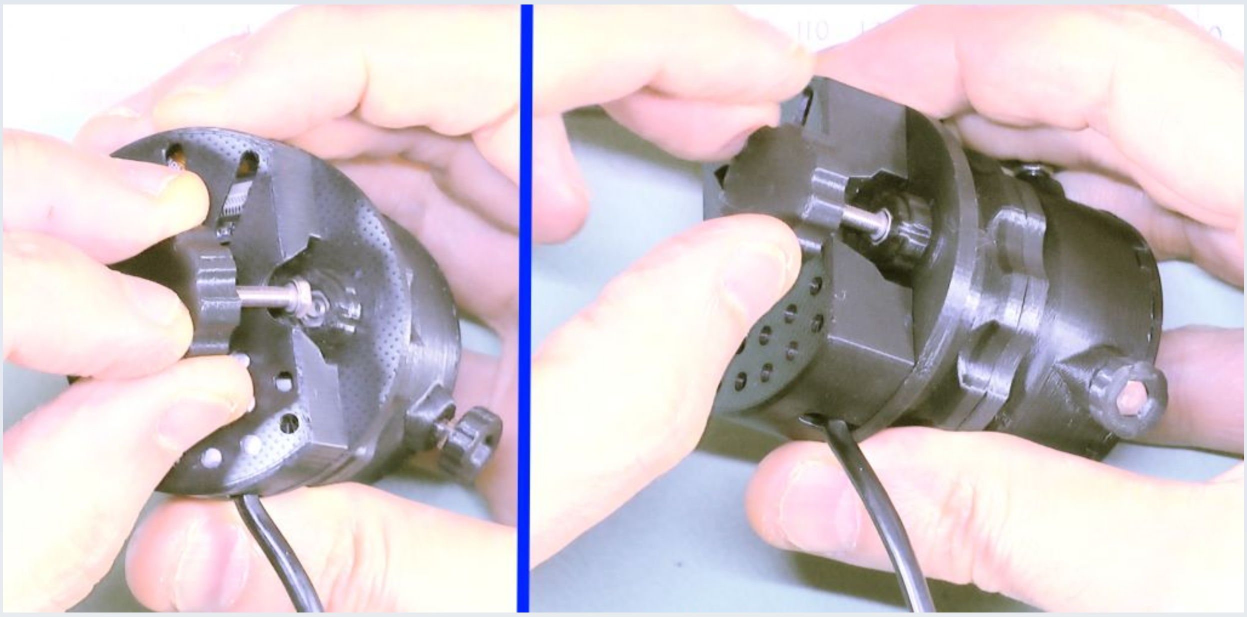


Figure 3.10 Using the adjust tool to fix the desired XY shift position of the adjustable mount. Left: Mate the hexagonal part of the tool into to the hexagonal socket on the thumbscrew nut. Right: Turn the tool to tighten or loosen the grip of the thumbscrew nut. Take care to avoid over-tightening or the tool may be damaged.

Eyepiece clamp

The eyepiece clamp is only required when using the camera with an eyepiece – you should not fit it if you are using the camera for ordinary or macro photography with its own in-built lens because it will obstruct part of the peripheral field of view.

To fit the eyepiece clamp first thread the lock nut onto it (item 5 in figure 1.1). For use with some eyepieces this lock nut needs to be positioned near the top of the thread (e.g. for use with some modern Leica eyepieces) and for others it should be positioned right down as far as it will go (see figure 3.11). The position of this lock nut determines the minimum distance of the sensor from the eyepiece and different eyepieces have different ‘eye-point’ distances (eye-relief distances) so this lock nut allows to adjust for those differences.

After this, screw the clamp with its lock nut onto the mount on the camera (figure 3.12).

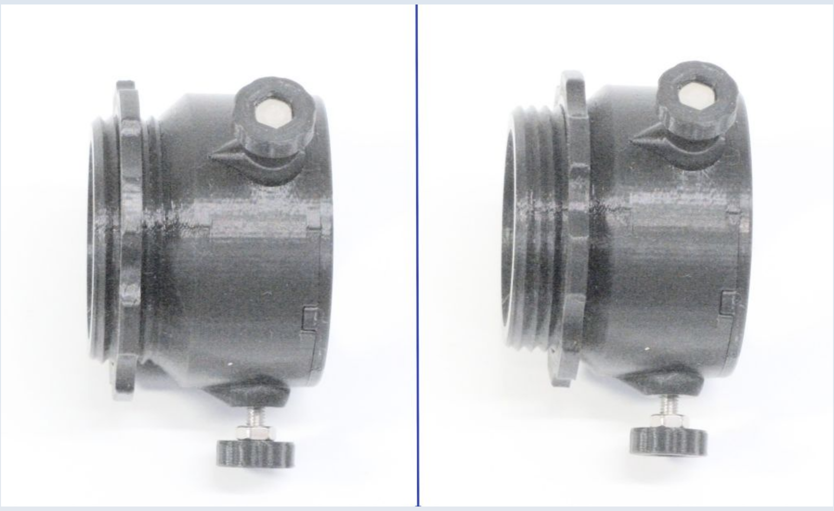


Figure 3.11 The eyepiece clamp and lock nut. You can adjust for the different eye-point / eye relief distances of various eyepieces by means of positioning the locknut at different points along the connecting thread of the eyepiece clamp. *Left*: The lock nut is near the end for eyepieces with longer eye relief (note there are about 2 full threads between the lock nut and the base of the threaded region). *Right*: The lock nut is right down the thread for those with shorter eye relief (i.e. 0 threads between the lock nut and the base of the threaded region). This adjustment cannot compensate for eyepieces with very short eye relief and such eyepieces will require a customised adapter to be designed and printed. Such customisations are possible due to the 3D printed nature of the clamp but will need to be done by the user – custom clamp are not supplied with the AF51 as sold by OptArc. For eyepieces with very long eye relief such that even the situation shown on the left does not give sufficient distance, the clamp can be positioned higher up the eyepiece to increase its range.

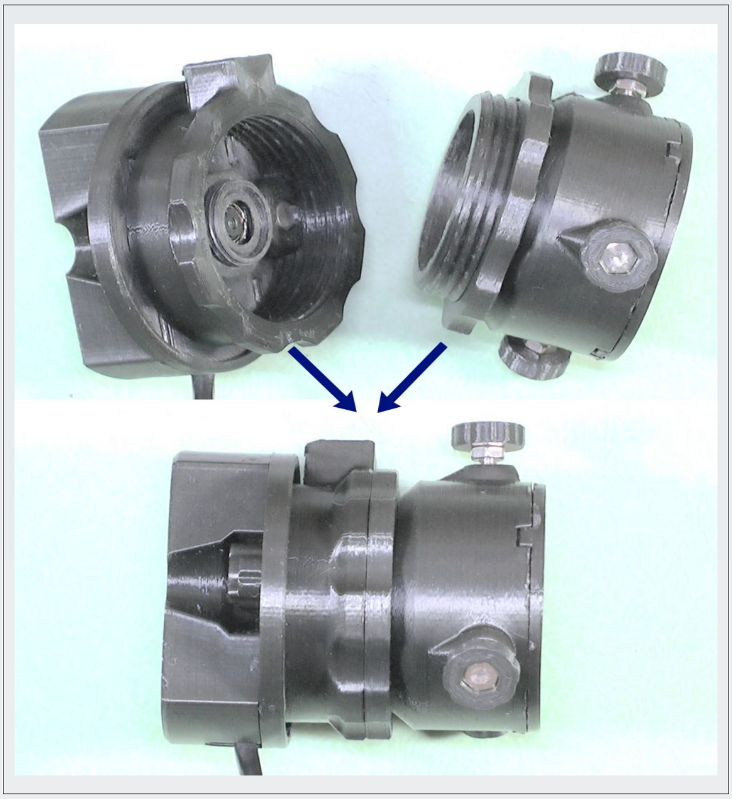


Figure 3.12 Screw the eyepiece clamp onto the adjustable mount as shown.

The clamp mechanism itself has three thumbscrews that push against three specially designed flexure areas of a thin internal plastic sheath (figure 3.13). These flexure areas are deigned to deform inwards to form 'folds' of plastic that grip the eyepiece by friction.

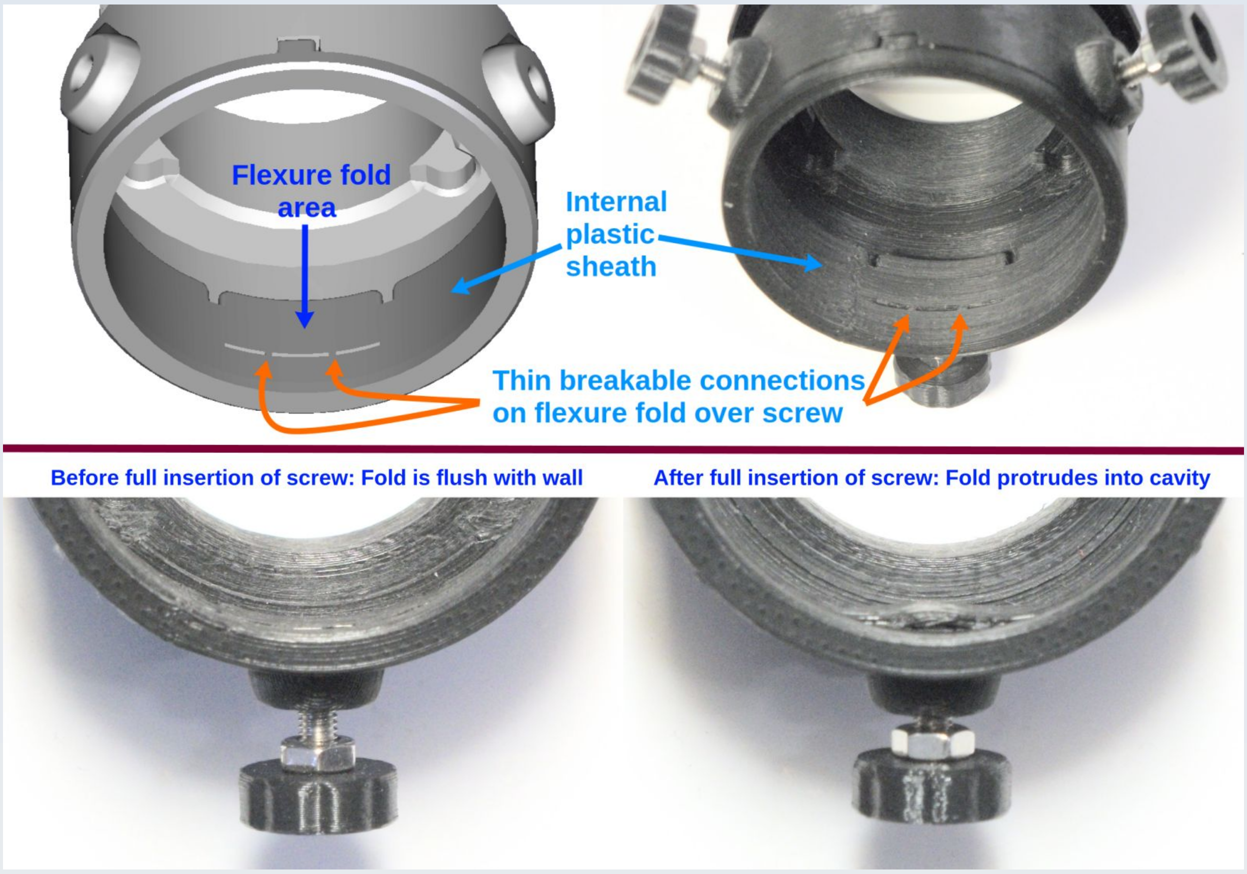

Flexure fold area
Internal plastic sheath
Thin breakable connections on flexure fold over screw
Before full insertion of screw: Fold is flush with wall
After full insertion of screw: Fold protrudes into cavity

Figure 3.13 Inside the eyepiece clamp. *Top*: The plastic sheath that overlies the three thumbscrews has a flexure fold over each screw with tiny plastic connections that are designed to break when the screw is sufficiently inserted (CAD model is shown left, actual clamp is shown on the right). *Bottom Left*: Before the connections break the flexure folds are flat against the inside of the clamp giving maximum clearance. *Bottom Right*: After the connections break there will always be some residual in-folding of the flexure giving narrower clearance.

When this clamp is brand new (freshly printed) the flexure folds of the inner sheath will be flush against the walls of the clamp (figure 3.12, bottom left) but with use these folds flex inwards and the tiny connections of the folds in the sheath ***are designed to break*** to allow maximum flexing to hold eyepieces of up to to 2.5 mm variation in diameter. Once those tiny connections are broken the indented folds of the sheath will never return fully flat against the clamp (figure 3.13, bottom right) – ***this is normal for the design of the clamp and is not a fault***.

This sheath is designed to protect the outer casing of the eyepiece from being damaged by the bare metal of the screws. **This protection is not absolute and some scratches or damage to the outer casing or its coating / paint-work may still occur due to the abrasive effect of the hard plastic of these folds** so the user must be aware of this and take whatever additional precautions they see fit to protect the casing of their eyepieces (such as applying some tape to the outer casing of the eyepiece before inserting it – although removing sticky residue left behind by some tapes may also cause damage to paint or other coatings so this risk is never fully zero).

If your eyepiece diameter is so big that the clamp will tighten against it without sufficient deformation to break the thin flexure fold connections then very little lasting deformation of the flexure folds in the clamp sheath will occur. However, when using the clamp mechanism after those flexure connections have broken (and so the flexure folds of the sheath protrude into the clamp), before inserting any eyepiece unscrew the three clamp screws and push the indented parts of the internal sheath folds up against the wall of the clamp with a finger or blunt instrument to achieve maximum clearance room inside prior to fitting an eyepiece. **Never use an instrument that could damage the plastic sheath (like a screwdriver or other metal instrument) to push these folds back because these can wear away or tear the protective parts of the thin plastic sheath that lie over the metal screws and expose the metal screws directly to your eyepiece casing, at which point damage to the casing will definitely result if you continue to use the clamp**. If the inner sheath does get damaged for any reason do not use it. Print and fit a new sheath using the freely available design files for this part of the camera (see below). One of the advantages of using the AF51 is that these parts can be relatively easily and cheaply replaced as required through 3D printing (if you don't have a 3D printer, there are online 3D print services that can be used).

I have made the CAD files for this clamp system open source. This means that you can modify the clamp design and print your own version to fit some eyepieces that will not fit with the standard clamp supplied with the AF51 or to replace worn or broken parts. See: https://github.com/TadPath/PUMA for the CAD files and keep an eye on the PUMA Microscope YouTube channel for instructional and demonstration videos that may help in this regard.

Once the inner sheath is prepared by pushing back any protruding folds, carefully insert your eyepiece about 20 mm into the sheath till it gets stopped by the inner ledge of the clamp (do not insert it any further than this even if it does not get stopped by the ledge) – see figure 3.14.

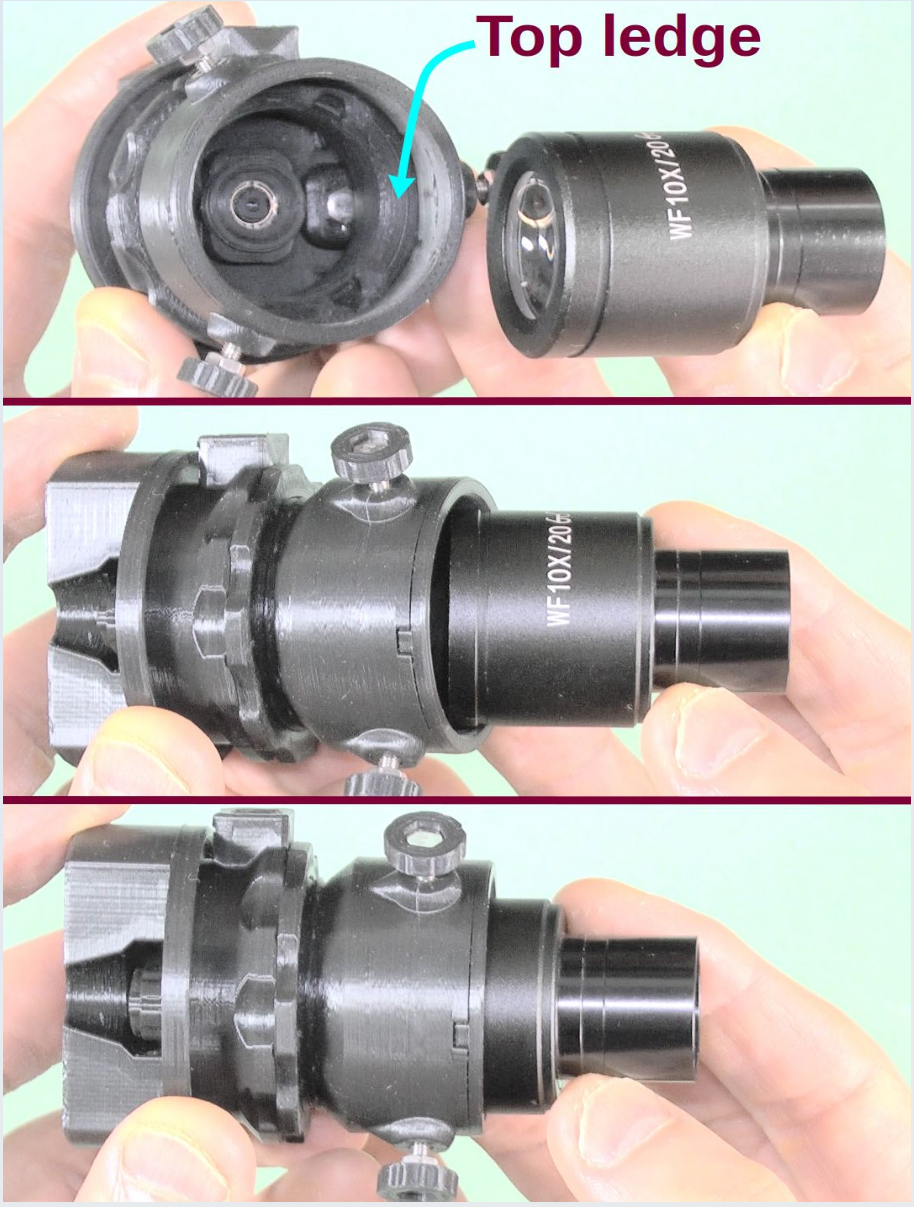


Figure 3.14. Insert the eyepiece body into the clamp till the top ledge of the clamp rests on the top of the eyepiece. This is the default position before starting to make adjustments for best image quality. Once in place tighten the clamp screws to hold the eyepiece steady but take care not to over-tighten the screws to minimise risk of damage to the body of the eyepiece. For some eyepieces with very long eye relief you will need to let the clamp 'ride high' on the body of the eyepiece so that the top ledge of the clamp no longer rests on the eyepiece. See text for details.

While the thumbscrews remain loose you should have some angular adjustment of the camera on the eyepiece as well as the ability to slip the clamp up or down on the eyepiece (see figure 3.15). All these adjustments together with the simple XY adjustment shown in figure 3.9 allow you to get the image of the eyepiece field of view centred in the camera field as well as achieve the correct eye point distance for for your eyepiece for optimally even illumination (see chapter 5 for more detail on how to do that).

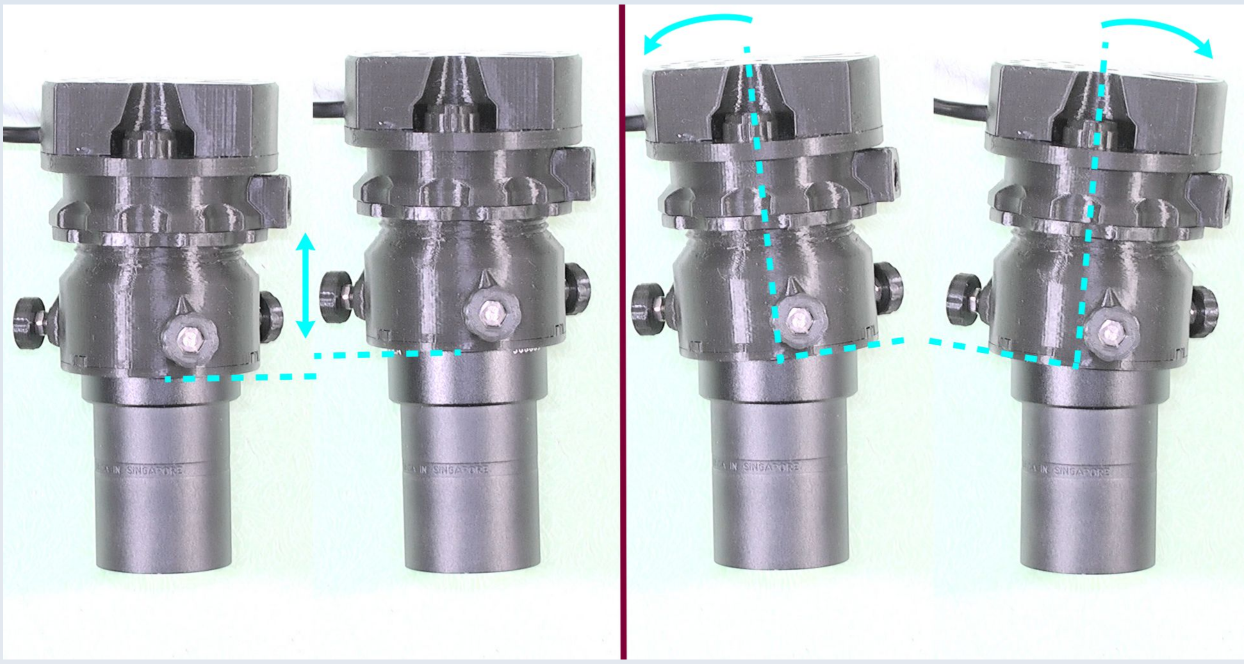


Figure 3.15 Additional adjustments made before the clamp screws are fully tightened. *Left*: Additional height can be gained by slipping the camera clamp up the body of the eyepiece for those eyepieces (such as the Leica L Plan shown here) which have a very long eye relief – more than can be accommodated by the lock nut adjustment of the clamp alone. *Right*: You can also tilt the camera on the eyepiece, exaggerated examples of which are shown here (it should never be necessary to use this much tilt in practice). Tilt can occur in all axes not just the one shown in this figure. If your eyepiece has a very wide body so that it only just fits in the clamp then the amount of tilt available will also be very restricted.

Now gently tighten the three thumbscrews till the eyepiece is held firmly in position but **do not over-tighten the thumbscrews or damage to the eyepiece casing or coatings may result. A sudden jolt to the camera while it is mounted on an eyepiece may also damage the eyepiece casing or coating.** These risks can never be fully removed so take care to avoid such incidents.

Tripod attachment

The translation adjustable mount has a female tripod thread socket (illustrated in figures 1.1, 2.2 and 3.9). This is designed to accept a standard 1/4-20 UNC threaded tripod screw. As most tripod screws are metal and this thread socket is made of 3D printed plastic **take care not to cross the threads or over-tighten when fitting the camera to a tripod or you may damage this thread socket**.

4. Additional hardware and software requirements

Hardware

The AF51 camera has no internal video or photo recording system and no internal storage or memory card slot to store recorded video or photos. All recording and storage must be done by an external host computer (not supplied).

Examples of types of external host computer that may be used with the AF51 are a desktop, laptop or tablet computer or a smart phone. Such devices must have a USB hub capable of connecting to the male type A USB connector of the camera (this may require an adaptor plug – not supplied with the camera).

The operating system of the host computer must be able to can handle communications with video devices that use the USB Video Class (UVC) protocol and run software that can use the UVC protocol to control UVC-compatible video devices. Such operating systems include Linux, macOS, MS Windows (Vista, 7, 8, 10, 11) and Android (4.0 and later).

The AF51 camera will not work with iOS (Apple iPhones and iPads) without some additional specialist interface electronics, the availability, quality and compatibility of which I have not tested. **Therefore I do not recommend using the AF51 with Apple iOS devices**.

The host computer you use must be able to supply the 500 mW of USB power required to power the camera.

Software

No separate drivers are required to use the AF51 camera because the UVC protocol built into the host computer's OS acts as a generic driver. For this reason the AF51 does not come with a driver disk or driver download link.

Software development kit (SDK) and custom programming

You can control the AF51 camera from your own custom software using freely available software development kits for Linux, Windows and Mac OS.

In Linux the v4l2 application programming interface (API) act as an SDK allowing you to control all functions of the camera described above.

For Microsoft Windows the SDK for Windows Vista and Windows 7.1 is freely available for download from Microsoft and contains example programs such as AmCap. These use the DirectShow API to control all the features of the camera described in this manual.

Although for older versions of Windows they will still compile and work well under Windows 10. A link and some compiling instructions for this SDK are given in the section on AmCap below. There is a newer alternative API for Windows called Windows Media Foundation which in theory should also be able to run the camera but I have no practical experience of using this at the time of writing this manual. The open source program 'Webcamoid' described below has a Windows Media Foundation option and this appears to work well.

I have no practical experience of macOS programming but an analogous SDK API for video devices on Apple computers is the 'AVFoundation' API. The Webcamoid software described below uses AVFoundation for its macOS support.

For those who are not into programming their own control software I present a selection of four ready-to-use software packages below. These not only let you use the AF51 but are also open source programs so they can serve as a guide and example to using the various APIs discussed above if you do intend to start writing your own programs eventually.

PARD Capture for Linux

I developed the free and open source ***PARD Capture*** software and optimised it to work with OptArc cameras including the AF51. It is released under the terms of the GNU GPL v.3

PARD Capture will be described in more detail in a separate chapter. It is designed primarily for scientific still image capture (including time lapse sequences and low light level work) and real-time video previewing but it is *not* suitable for real-time video capture.

PARD Capture allows full control of all the AF51 features described in the previous chapter as well as providing additional functionality like real-time frame integrating preview for low light applications, dark field and flat field correction, the ability to save images as raw doubles or floating point FITS files, multi-frame averaging, the ability to read out just the Y component from the YUYV stream, etc.

At the time of writing this manual, PARD Capture is only available for Linux and is only available as source code that you will need to compile. For those using other operating systems or those who do not wish to compile source code, a small selection of other free software packages that you may find useful is listed below but bear in mind that there are many others which are not mentioned below but which may be even better for your use case so some research on your part is advised. The main qualification that any software package must have to be able to work with the AF51 camera is that the software must be able to interface with UVC-compliant video devices.

The project page link is: https://github.com/TadPath/PARDUS/tree/main/PARD_Capture

Guvcview for Linux

Guvcview is a free and open source video and still frame capture program for Linux by Paulo Assis and licensed under the terms of the GNU GPL v.3.

It can control all the functions of the AF51.

It can save video streams as well as still images.

It can use the uncompressed YUYV format of the camera.

It can save images in uncompressed formats as well as compressed formats. This includes saving as ‘raw’. However, unlike PARD Capture there is no option to save just the Y channel component. You will need to figure out how to extract that from the ‘raw’ format if you need it (and it ***is*** needed for high quality imaging at low light levels with the AF51).

It comes as an installable package for various forms of Linux so does not need to be compiled by the user (although on some versions of Linux you may need to compile it if an install package is not available).

It is only for Linux.

The project page and source code link is: https://sourceforge.net/projects/guvcview/

Microsoft AmCap and SDK for Windows

AmCap is open source software by Microsoft. It is distributed under the MIT license.

It is only for Microsoft Windows. It is legacy code that is for Windows 7.1 and Windows Vista but it can also work on Windows 10.

It can control all the functions of the AF51. The ‘Exposure, Dynamic Framerate’ control is called the 'Low Light compensation' control in AmCap.

It can only save video, and then only in AVI format. It does not save stills and you cannot save only the Y channel component.

It can use the uncompressed YUYV format of the camera.

The source code GitHub page is here: https://github.com/microsoft/Windows-classic-samples/blob/main/Samples/Win7Samples/multimedia/directshow/capture/amcap/amcap.cpp

I recommend you compile the program yourself from the official Microsoft source code and I give some instructions on how I did that below. If you can’t do this then **be aware that many sites offer the AmCap executable to download but**

some of these downloads and installation packages may also contain malware / viruses / adware so this route is not advisable.

One of the advantages of compiling this yourself – if you have programming skills – is that you will have the source code for the program which will give you a good example of how to use the MS DirectShow API for controlling a camera (including the AF51) in your own programs so you can eventually write your own software based on some variation of this code to make up for the deficiencies of AmCap (e.g. its inability to save still images) as well as add more useful features such as the low light level imaging features found in the PARD Capture software for Linux (multi-frame integration, frame averaging, saving raw data in floating point formats, saving the Y-only data, etc.).

On Windows 10 I compiled AmCap.exe like this:

1. Download MS software development kit (SDK) for Windows 7.1 / Vista from here:

https://msdn.microsoft.com/windowsvista/bb980924.aspx

The SDK contains many components and examples, one of which is AmCap.

2. Download Free Community Edition of MS Visual Studio from Microsoft (you will need a Microsoft account to do so).

3. Start (or re-start) Visual Studio as an Administrator.

In the following instructions when you first open a solution in MS Visual Studio it may ask you to do a one-way upgrade of the solution files to the current version of MS Visual Studio – do that when prompted.

4. File-> Open -> Project/Solution

Navigate to the SDK folder for the 'Base classes' solution in the SDK e.g.:

```
C:\Program Files\Microsoft SDKs\Windows\v7.1\Samples\multimedia\directshow\baseclasses
```

5. Select the type of build you want to create (Debug or Release) from the project configuration drop-down selector menu near the top of the MS Visual Studio window. The default configuration when you first download and start MS Visual Studio will be ‘Debug’.

6. Build this solution (Build -> Build solution).

7. Once successfully built, open the ampcap example solution (File -> Open -> Project/Solution) which will be located in the SDK folder e.g.:

```
C:\Program Files\Microsoft SDKs\Windows\v7.1\Samples\multimedia\
directshow\capture\amcap
```

8. Select the type of build you want to create (Debug or Release - use the same one you used for building the baseclasses above) then build this solution (Build -> Build solution)

9. If it worked you will have a working copy of AmCap.exe in the location appropriate to the type of build you made (debug or release) e.g. in

```
C:\Program Files\Microsoft SDKs\Windows\v7.1\Samples\multimedia\
directshow\capture\amcap\x64\Debug\AmCap.exe
```

or

```
C:\Program Files\Microsoft SDKs\Windows\v7.1\Samples\multimedia\
directshow\capture\amcap\x64\Release\AmCap.exe
```

10. Connect your AF51 camera to the computer then double-click the AmCap.exe file.

Webcamoid for macOS, Android and MS Windows

Webcamoid is free and open source software for Linux, Android, macOS and Windows by Gonzalo Exequiel Pedone and released under the terms of the GNU GPL v.3.

The MS Windows version of this software can control most ***but not all*** the functions of the AF51. In v.9 of this software (the latest version tested when writing this manual) there is no separate control for ‘Exposure, Dynamic Framerate’ or for the ‘Power Line Frequency’. Other controls of the camera are present and work as expected. Note this was only tested on the MS Windows version of the program so availability of controls under Linux and macOS may be different.

It can save video streams as well as still images.

It can use the uncompressed YUYV format of the camera.

It can save images in uncompressed formats as well as compressed formats but you cannot save only the Y component of the YUYV stream.

It comes as an installable package for various forms of Linux as well as Mac OS X and MS Windows so does not need to be compiled by the user (although on some versions of Linux you may need to compile it if an install package is not available).

It is available for various forms of Linux (64 bit), macOS (10.14 or later, 64 bit) and MS Windows (Windows 8 or higher, 32 or 64 bit). The Android port of Webcamoid is relatively recent. Check the project pages for the latest on Android compatibility.

The project page and source code links can be found here: https://webcamoid.github.io/

Apple macOS and iOS

The camera should work on macOS (previously called OS X or Mac OS X) using the Webcamoid software described above and other software capable of running UVC video devices. You should also be able to write your own custom software for controlling the AF51 camera using Apple's 'AVFoundation' API (see the source code of Webcamoid for examples of how to use this API).

However the AF51 has not been tested by me on Apple computers at the time of making this manual so I cannot confirm if all the AF51 functions described in this manual are available on Apple computers.

Apple iOS devices will not operate the AF51 camera without additional hardware interface electronics (that I have not tested) so currently I do not recommend use with Apple iOS devices.

5. Fitting the AF51 to an eyepiece

The basics of inserting an eyepiece into the camera's eyepiece clamp were already given in chapter 3. This chapter will give detailed information about fitting the camera to various kinds of eyepiece and explain how adjust the camera on an eyepiece in order to get an optimal image.

I have tested the AF51 and its standard eyepiece clamp with 13 different eyepieces that span the generations from vintage classics to modern examples, mostly for microscopes but also some 1.25 inch barrel eyepieces for telescopes (figure 5.1).

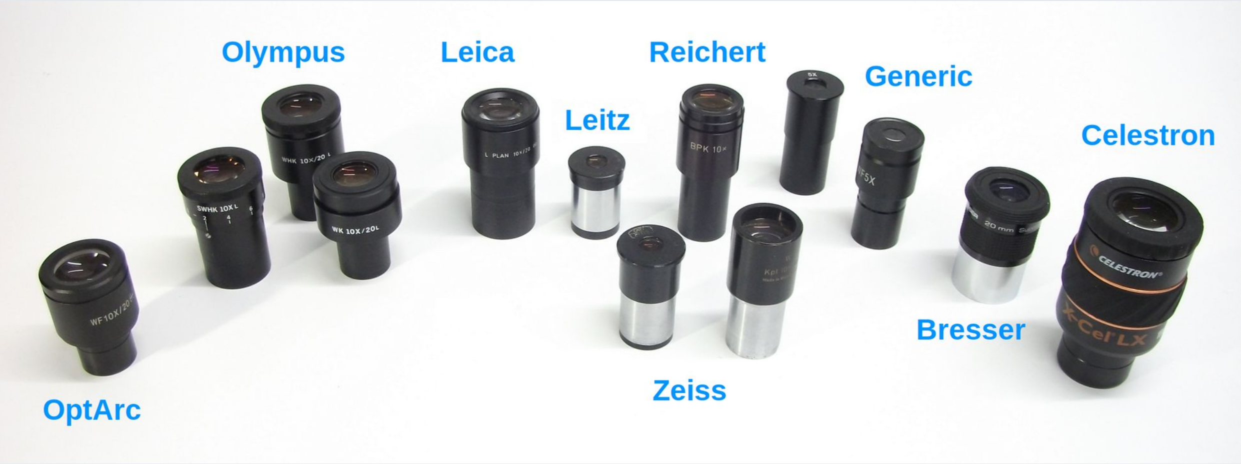


Figure 5.1 The 13 eyepieces tested with the AF51 and results of which are described in this chapter. Not all these will be able to fit in the standard eyepiece clamp provided with the AF51.

Not all these eyepieces can be used with the standard AF51 eyepiece clamp and not all of them provide a field of view that fills the camera frame sufficiently to give a high resolution image. Details will be provided in this chapter. Of course I cannot test all possible eyepieces so I will list the eyepiece measurements alongside the results so the reader can use this data as a guide as to how their own eyepiece is likely to perform with the AF51 and its supplied clamp. In general, the results show that eyepieces with short eye relief and small diameter exit lens do not give a satisfactorily large projection of the image to make it useful for imaging with the AF51.

Eyepiece body diameters – limits of usability

The 'maximum outer diameter' measurements referred to in this section are the diameter of the part of the eyepiece that does NOT go into the microscope tube. This measurement is of the widest part of the eyepiece including any protruding knurled grip area and not just the plain casing of the eyepiece. This is the part of the eyepiece that goes into the eyepiece clamp of the AF51. Some examples are illustrated in figure 5.2. Note that this measurement only determines if the body of the eyepiece can fit and be held into the standard eyepiece clamp of the AF51. The fact that an eyepiece can fit into the clamp does not necessarily mean it will be useable with the AF51 camera with this clamp because that will also depend on the level of the eye point of the lens. This will be discussed in a later section.

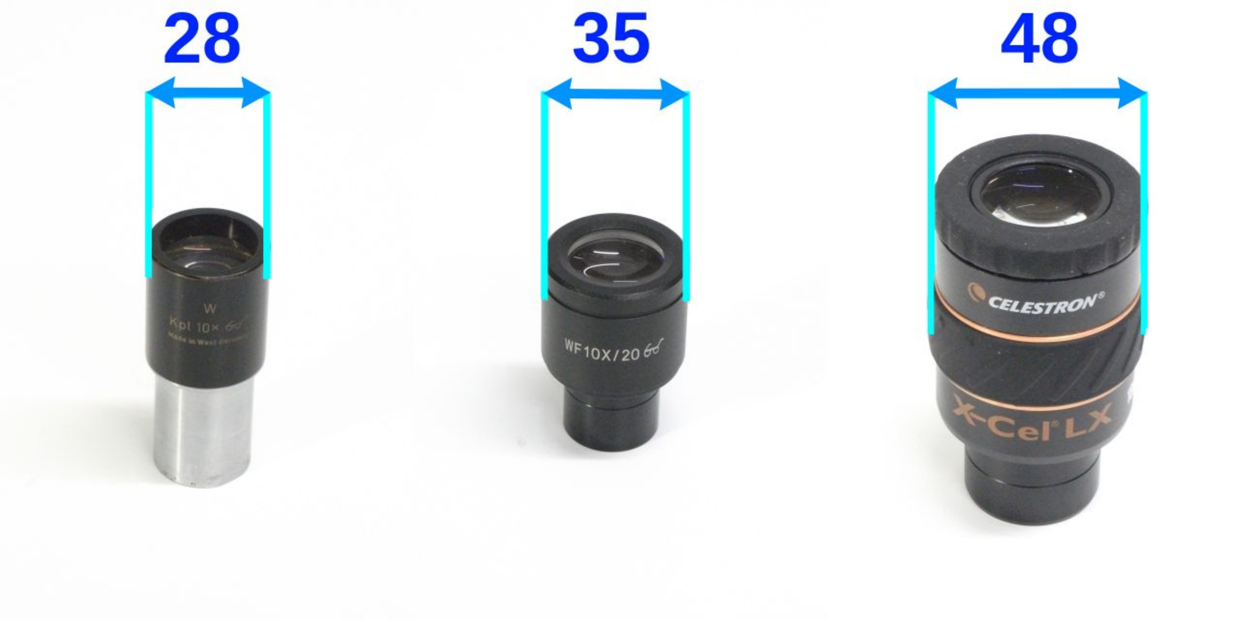


Figure 5.2 Examples of eyepiece maximum outer diameter (in mm) for three eyepieces. *Left*: Zeiss W Kpl x10, *Middle*: OptArc WF x10/20, *Right*: Celestron X-Cel LX 18 mm. Maximum body diameter measurements are shown above each eyepiece (units are mm). The Zeiss will be too small to fit the standard AF51 eyepiece clamp without a special adapter (not supplied). The Celestron will be too large to fit. See text for details.

The standard clamp mechanism of the AF51 was designed to work optimally for eyepieces with a maximum outer diameter of 35 mm to 36.5 mm. However it will work with eyepieces that have a maximum outer diameter in the range 33 mm to 37.5 mm (inclusive) but the upper end of that scale needs special attention (detailed below). For eyepieces with maximum outer diameters outside this range, custom adapters need to be used (these adaptors are not supplied with the AF51).

Examples of eyepieces with a maximum outer diameter in the range 35 mm to 36 mm include the OptArc x10/20 WF, the Leica L Plan x10/20 (including the ‘M’ variant), the Olympus SWHK 10xL and Olympus WHK 10x/20 L.

Eyepieces with a maximum diameter in this range should be useable even when the clamp sheath fold connections have broken and the folds protrude into the clamp (provided you insert the eyepiece after pushing those folds back manually as described in chapter 3 – see figure 3.13).

Eyepieces with a larger maximum diameter up to 37.5 mm (like the Olympus WK 10x/20L series) can just fit when the clamp sheath is new (before breakage of the flexure fold connections in the sheath). Once the clamp sheath flexure fold connections have been broken such larger eyepieces will not fit without excessive pressure that brings with it a high risk of damage to their casing finish so this should not be attempted. Furthermore, tight-fitting eyepieces will have very limited angular adjustment room (see figure 3.15) so may not be able to project a well-centred, evenly illuminated image (although my tests with the Olympus WK 10x/20L-H gave very good results).

The lower limit of usability with the standard eyepiece clamp of the AF51 is an eyepiece body diameter of about 33 mm. Smaller than this and the clamp will not be able to grip the body of the eyepiece.

Some eyepieces have a significantly narrower diameter e.g. some vintage eyepieces from Zeiss (e.g. 28 mm diameter for the Zeiss W Kpl x10) and Reichert (e.g. 29 mm diameter for the BPK 10x) as well as some modern cheap generic eyepieces. These are too narrow to be gripped by the indentations of the inner sheath of the AF51 standard eyepiece clamp even when the 3 clamp screws are fully tightened.

For this reason such ‘thin’ eyepieces require an optional narrowing adaptor insert (figure 5.3). **This adaptor is not provided with the AF51** but the CAD file for it is provided on the PUMA Microscope GitHub page. To use it simply orientate it with the 3 lugs on the adaptor matching the 3 recesses in the clamp and insert the adaptor all the way in till the lugs mate into the recesses.

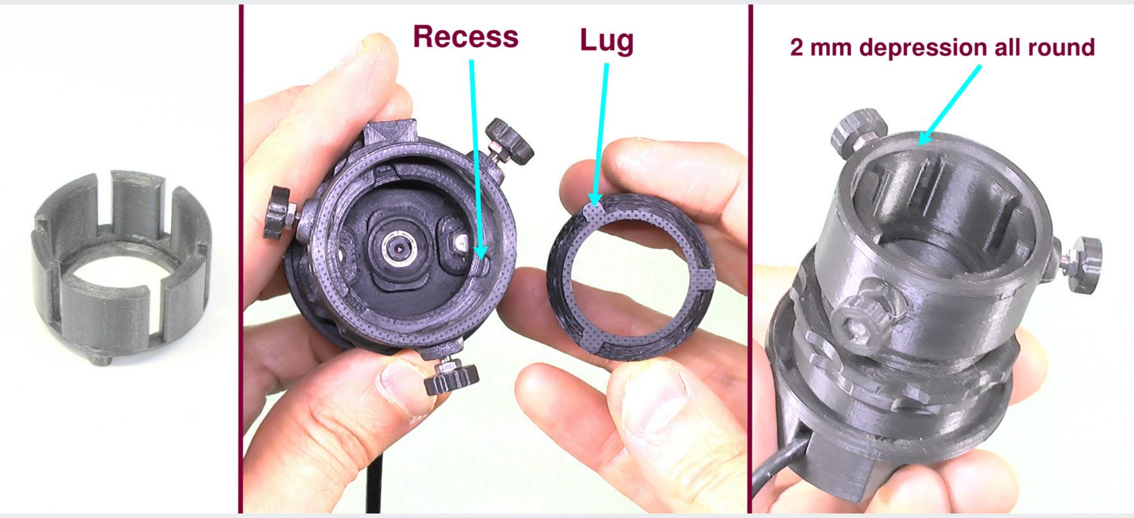


Figure 5.3 The optional thin eyepiece adaptor. *Left*: The adaptor with its vertical flexible flaps. *Middle*: to fit is align the 3 lugs with the 3 recesses in the clap and insert it all the way in so that these mate. *Right*: A correctly fitted adaptor will lie 2 mm below the rim of the clamp evenly all around.

If this is done correctly the top of the adaptor will lie about 2 mm below the rim of the clamp all round. Now screw in the clamp screws a little till the flexure folds of the clamp begin to squeeze the flaps of the adaptor. Insert your eyepiece at this stage and continue to tighten the clamp screws till the eyepiece is held in place.

Eyepiece eye relief – limits of usability

Even if an eyepiece has a diameter than can fit the standard AF51 clamp (with or without the thin adaptor) it must also have a suitable level of eye relief in order to be used with this clamp.

In the tests that I have done with the 13 eyepieces shown in figure 5.1, all the eyepieces can be imaged properly by the camera sensor even if they have a very short eye relief but not all of them can be so imaged with the standard AF51 eyepiece clamp. This is because eyepieces with a very short eye relief cannot be brought close enough to the camera lens to get the full field of view into the camera with the standard clamp. For those eyepieces with short eye relief, therefore, some custom adaptor will again need to be made or acquired by the user (they are not supplied with the camera).

To give you an idea of whether the standard clamp will work fully with a particular eyepiece I have tabulated my results with the 13 eyepieces tested. Manufacturers use various methods to measure eye relief distance and not all eyepiece models quote this eye relief value. For that reason, instead of trying to list a manufacturer's listed eye relief value I made my own surrogate measurement, which I call the 'eye point' distance, defined as the distance away from the top-most solid part of the eyepiece where the disc of light coming from the eyepiece is the smallest when a specimen is in focus on the stage with a x40 objective (figure 5.4).

All measurements were made on a PUMA microscope with its Abbe condenser and Kohler illuminator in situ. I stress this setup because I have found that the same measurement technique made with the same eyepiece on a different microscope gave slightly different results for this eye point measure.

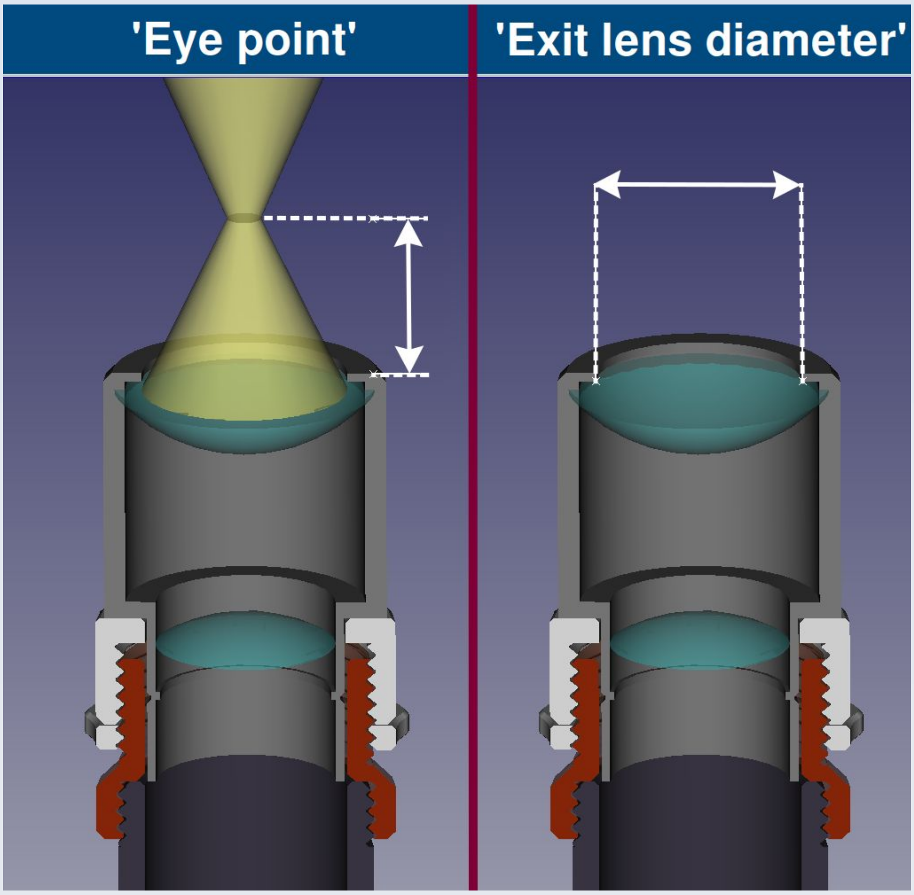


Figure 5.4 Cut-away diagram of the top part of the PUMA microscope with simplified schematic of an eyepiece in situ. *Left*: The ‘eye point’ distance used in table 5.1. *Right*: The ‘exit lens diameter’ used in table 5.1

For example, using my method the ‘eye point’ of a Leica L Plan eyepiece came to 23 mm as measured on the PUMA microscope but 20 mm as measured with the same technique on the Leica DMLS microscope. Also, the ‘eye point’ on the PUMA microscope of an Olympus SWHK 10xL eyepiece came to 17 mm but when measured on an Olympus BH2 BHS microscope it came to 15 mm. I have an actual manufacturer ‘eye relief’ figure for the the two astronomical eyepieces tested in this group – the Bresser 20 mm SPL and the Celestron X-Cel LX 18 mm – both of which are claimed by the manufacturer to have an eye relief of 16 mm. My measurement of eye point gave a value of 15 mm for the Bresser and 18 mm for the Celestron. This difference may be accounted for by the fact that the level of the focal plane of astronomical eyepieces are not standardised like microscope eyepieces to be exactly 10 mm below the shoulder of the eyepiece.

For all these reasons my ‘eye point’ measure does not equate to eye relief. However, this eye point measure did correlate well with the actual eye relief experienced with each eyepiece in that those with a longer eye point had longer eye relief in practice.

The results of my tests are shown in table 5.1.

Eyepiece	Max. Diam	Eye point	Exit lens diam	Diam. OK	Eye point OK	Field % pixels	Field MP	Full circle field	Tested on	Comments
AE 5x Huygens	28 – 25	13	10.5	No	No	12.24	0.62	Yes	PUMA	Prominent lip
Leitz Wetzlar Periplan NF x10	26 – 25	11	10	No	No	40.3	2.03	Yes	PUMA + Zeiss	Eye-point OK with clamp locknut removed but due to very short eyepiece body (7 mm) the PUMA ocular locknut gets in the way of the AF51 clamp.
AE 5x WF	28	13	10.5	No	No	11.46	0.58	Yes	PUMA	
OptArc x10/20 WF	35	17	22	Yes	Yes	49.2	2.48	Yes	PUMA + Zeiss	Locknut 1.5 threads up
Bresser 20 mm SPL 52 degree FOV	33	15	18	No	Yes	54.86	2.76	Yes	PUMA	Locknut 0.5 threads up. Diameter too thin for standard clamp.
Celestron X-Cel LX 18 mm 60 degree FOV	48	18	26	No	Yes	77.01	3.88	No	PUMA	
Leica L Plan	36 – 35	23	20.5	Yes	Yes	48.1	2.42	Yes	PUMA + BH2 + Leica DMLS	Locknut as high as possible and clamp riding high on the eyepiece
Olympus SWHK 10xL	36	17	22	Yes	Yes	76.75	3.87	No	PUMA + BH2	Locknut at 1 full thread
Olympus WK 10x/20L	37.5	16	17.5	Yes	Yes	44.69	2.25	Yes	PUMA +BH2	Locknut very close to base.
Olympus WHK 10x/20 L	36	20	19	Yes	Yes	46.9	2.36	Yes	PUMA +BH2	2 threads
Zeiss W Kpl 10x	28	18.5	21	No	Yes	37.75	1.9	Yes	PUMA + Zeiss	2 threads with thin eyepiece adaptor
Reichert BPK x10	29	19	18.5	No	Yes	38	1.91	Yes	PUMA + Zeiss	2 threads with thin eyepiece adaptor
Zeiss Kpl 8x	28 – 25	10	9.5	No	No	24.9	1.26	Yes	PUMA + Zeiss	

Table 5.1 Results of imaging with 13 different eyepieces using the AF51 camera. Not all the eyepieces could be used with the standard eyepiece clamp. For those that could not I held the camera manually over the eyepiece at the correct height and orientation to take a sample photo for the purposes of measuring the proportion of image used by the field of view disc. The 'eye point' is my own measurement as described in the text and figure 5.4. 'AE' refers to low cost generic eyepieces from Ali Express.

Notes:

Where a range of maximum diameter is given in the table this reflects the fact that there is a prominent rim or ledge to the body of the eyepiece so the larger of the two measurements refers to the diameter of the ledge. The presence of a prominent ledge can make it difficult to mount in the standard eyepiece clamp of the AF51.

The 'Exit lens diameter' is the clear aperture of the final lens in the eyepiece nearest the user's eye (this is sometimes called the 'ocular lens' but that term also refers to the eyepiece as a whole in many contexts so I just use 'exit lens' – not to be confused with 'exit pupil' of the optical system).

'Diam. OK' means you can clamp this eyepiece in the standard AF51 eyepiece clamp without a custom or optional adaptor.

'Eye point OK' means the standard AF51 eyepiece adaptor can hold the eyepiece at the correct distance for optimum imaging of the field of view.

'Full circle field' described whether the image includes the whole round field of view. In the two most wide field eyepieces tested this is not the case because the width of the eyepiece field is wider than the width of the camera field of view.

All the eyepieces were tested on a PUMA microscope with some also tested on other microscopes (a Zeiss Standard, an Olympus BH2 BHS and a Leica DMLS). Field of view measurements were about the same regardless of which microscope was used (but eye point measure differed – only the eye point measure from the PUMA microscope is shown).

The number of threads stated in the comments refers to how high the eyepiece clamp lock nut needed to be to get optimal imaging and refers to the number of threads showing from from the base of the threaded part of the clamp to the lock nut (see figure 3.11).

'Field % pixels' and 'Field MP' refer to how much of the full 5 MP image field is occupied by the microscope FOV in the image. The range of useable pixels between these eyepieces is quite large (see figure 5.5). Some of the cheapest eyepieces, while useable for direct vision, give such small coverage in the AF51 camera field of view that the available pixels encompassed by the eyepiece field of view is very small (less than 1 MP).

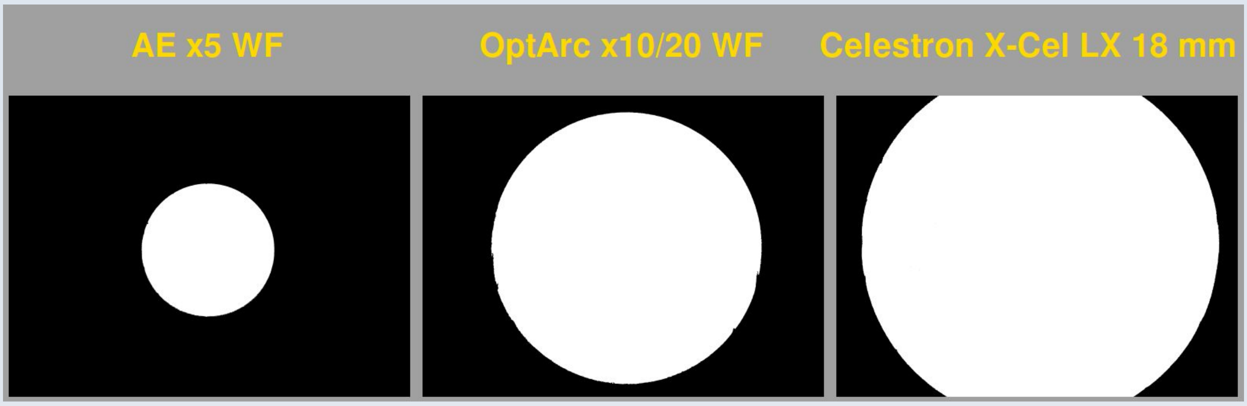


Figure 5.5 The range of field coverage by different eyepieces is very wide. *Left*: The low cost 'WF' x5 eyepiece from Ali Express only uses 0.58 MP of the full 5 MP frame. *Middle*: The OptArc x10/20 WF eyepiece fills almost the full height of the frame and uses 2.48 MP of the available 5MP. *Right*: The Celestron astronomical eyepiece has the widest field of view of all those tested – so wide that it overspills the vertical aspects of the frame of the camera to use 3.88 MP of the available 5 MP at the expense of not showing the full field of view. For some uses this will be preferable (e.g. if you want to crop the field of view to a full rectangle). Of these three, only the OptArc eyepiece will be useable with the standard eyepiece clamp that comes with the AF51 camera and the Celestron eyepiece will also need a custom adapter to fit it to the microscope (most biological microscopes do not use a 1.25 inch barrel). Details of field of view usage by the other eyepieces tested are given in table 5.1

Adjusting on an eyepiece to get a good image

Preliminary steps and aims

The first thing to do is ensure the microscope is on and, looking down the scope by eye using the eyepiece you intend to use for photography, you see the whole field of view of the scope is evenly illuminated and you can see a specimen in focus.

Next ensure the camera is on and you can see a live image preview (the MJPEG image stream setting is useful for this purpose because it will usually give a high frame rate regardless of the resolution). Ensure that auto focus is OFF and that the manual focus is set to infinity (the default value of 68 should suffice for that).

Now slip the camera over the eyepiece so that the eyepiece is fitted into the clamp of the AF51 (as described in chapter 3 in the section 'Eyepieces clamp').

At this point you will either see no image, a partial round field of view or the whole image field of view. You will most likely need to perform some centration adjustments before tightening the clamp screws in order to see an image (at all) and to get the best quality image. These adjustments are necessary to align the optical axis of the scope with the optical axis of the AF51 camera's lens system and to ensure the exit pupil of the scope is at the correct height for the camera so the camera sees the full field of view without vignetting. Failure to make these adjustments correctly can result in no image being seen at all or a very unevenly illuminated or small image being seen.

Types of adjustment available

The kind of adjustments that can be made are:

1. XY shift using the thumbscrews on the translation adjustable mount (see figure 3.9)

2. Angular tilt of the clamp on the eyepiece (see figure 3.15, right)

3. Height adjustment using either the clamp lock nut alone or accompanied by slipping the clamp up on the body of the eyepiece (see figures 3.11 and 3.15, left).

Centring the optical axis

The first thing to do is centre circular eyepiece field of view in the rectangular camera field of view. This is done with adjustments 1 and 2 described above but because adjustment 1 (XY shift) is already set at a nominal central position you should begin by making angular adjustments till you catch the optical axis in the camera's field of view then make small corrective adjustments iterative in both shift and angle till the desired result in achieved (see figure 5.6).

You should have the camera set to auto exposure mode and use a high frame rate preview live image while doing this adjustment. This is because you need real-time visual feedback and the light levels entering the camera can change radically when off axis compared to when on axis. Without the auto-exposure the camera preview image may look completely black or completely white and so you will not be able to just alignment.

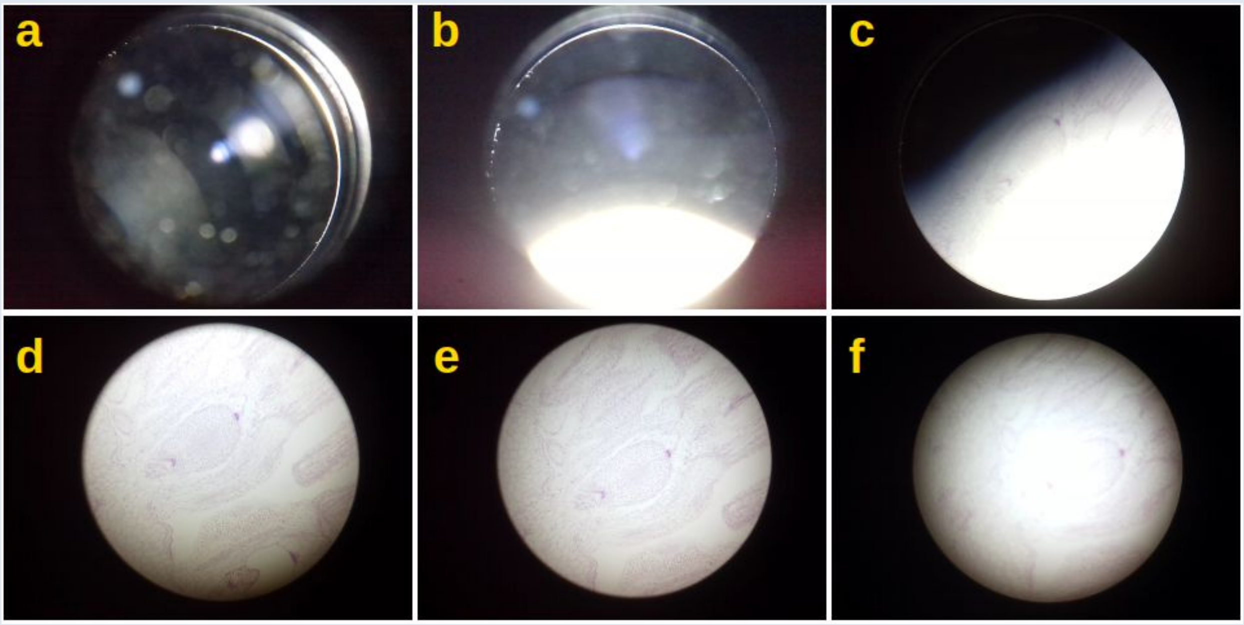


Figure 5.6 Catching the optical axis and centring the field. a) When you first insert the camera you may see no image at all. The small blurry circles are dust specs on the eyepiece. b) and c) Adjusting the angle of the camera on the scope will begin to bring the optical axis into alignment. d) The optical axis is almost centred but the fact that the brightest spot is off-centre means there is still some adjustment to make. e) The optical axis is aligned (the brightest area is central) BUT the circular field of view disc is not centred in the camera's rectangular field of view. f) A few more small adjustments centres everything – the field of view disc is centred in the rectangular camera field and the central bright area is centred in the circular field of view. We can now progress to reducing the vignetting by height adjustments (see figure 5.7).

Do not be concerned about vignetting at this stage (the light being darker at the periphery than the centre) but do ensure that the brightest part of the image is well centred in the circular field of view. Also, ignore the position of the specimen under the microscope – you are only concerned about the patterns of illuminating light (in figure 5.6 I moved the specimen on the stage in between image captures so it will not look in the same position in each image – but that is irrelevant, as is any slight defocussing).

Angular adjustment tends to move the circular eyepiece field more than XY translational adjustment but too much adjustment of either angular or XY shift on their own will tend to loose the image as the optical axis of the scope and camera lens come out of alignment. These two adjustments must therefore be made incrementally with each other.

If the camera is extremely out of alignment you will loose image altogether so if you do not see an image at all you should adjust angle and XY shift till you 'catch' the image from the eyepiece. This can be quite tricky sometimes because there is a very narrow window of adjustment settings where the image can be seen.

Achieving the correct eye point level

Once the fields of view are centred you adjust the height of the camera above the eyepiece to achieve optimal field flatness (evenness of illumination) with a sharp boundary to the circular field of view. Some eyepieces give a softer boundary than others and some microscope illumination systems can never give a completely flat illumination but the point is you want to find the height that minimises any vignetting (figure 5.7).

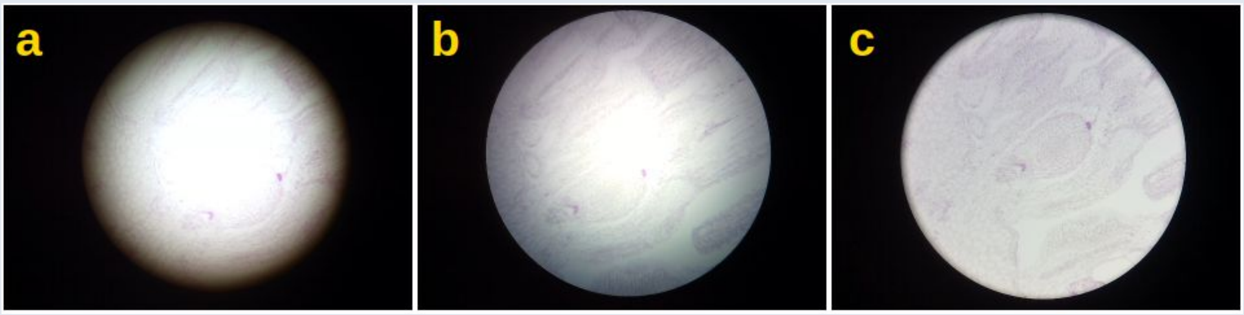


Figure 5.7 Adjusting the height of the camera above the eyepiece such as to provide the flattest illumination across the field. a) The camera is too high above the eyepiece. b) The camera is too close to the eyepiece. c) The camera is just at the optimal level. Note that while you make this adjustment you may need to also adjust the axis as well (as shown if figure 5.6) because while making these height adjustments you may throw the camera slightly off axis.

The pictures in figures 5.6 and 5.7 were taken using a PUMA microscope with Abbe condenser and Kohler illuminator using a x20 Olympus SPlan objective.

6. Introduction to PARD Capture

Correctly mounting the camera over the eyepiece (as described in chapters 3 and 5) is essential for taking good images but, in itself, is not sufficient. The AF51 camera has a range of settings (described in chapter 3) which need to be set appropriately to the type of image acquisition needed for different purposes. In addition there are particular image acquisition techniques, such as multi-frame averaging, dark current correction, flat field correction, etc., which need to be performed to get the best possible images for scientific applications.

Failure to select the correct camera settings and failure to adopt the correct image acquisition procedure can result in images of disappointing quality. Not all the software packages discussed in chapter 4 allow for optimal image capture for scientific processing. For example, none of them allow you to capture only the monochrome Y channel of the YUYV frame (and that is important to avoid artefacts that occur in the colour signal readout of the AF51 in low light conditions – see chapter 7 for more on that). The free and open source PARD Capture software, on the other hand, was specifically designed for optimal image acquisition for scientific image capture (including Y-only images) and this chapter will take you through the basics of its usage. Chapter 7 will give some practical tips for optimal quality image acquisition and show example images.

PARD Capture uses the v4l2 API to control the default video device on the system (/dev/video0) so it may work with many different cameras. However it was optimised to work with OptArc cameras so some other cameras may not be compatible. In order to be useable with PARD Capture a camera must be able to stream images in either MJPEG or YUYV (or both) image formats.

For more information about PARD Capture and to get the latest version of the source code, see the project GitHub page:

https://github.com/TadPath/PARDUS/tree/main/PARD_Capture

Compiling the software

PARD Capture comes as source code written in the C programming language. To compile it into an executable program you will need to ensure that you have first installed all the libraries it depends on (i.e. its dependencies). You will need the developer libs for v4l2, gtk3, libjpeg and libpng. The audio GUI signals require the PulseAudio libraries which may already come with your Linux as standard but, if not, you must install it. Table 6.1 shows the libraries you need to install with their names as used in Arch, Fedora, Debian and Ubuntu Linux distros:

Library	Arch	Ubuntu/Debian	Fedora
v4l2	v4l-utils	libv4l-dev	libv4l-devel
JPEG	libjpeg-turbo	libjpeg-dev	libjpeg-turbo-devel
GTK	gtk3	libgtk-3-dev	gtk3-devel
PNG	libpng	libpng-dev	libpng-devel
Pulse	libpulse	libpulse-dev	pulseaudio-libs-devel

Table 6.1 Dependencies for PARD Capture. Install these using your Linux installation method before compiling the PARD Capture software.

Note: Arch Linux installs the developer parts of a lib as standard with each lib. Other types of Linux need to specifically install the ‘dev’ or ‘devel’ versions of the lib.

Once all the dependencies are installed compile the PARD Capture code itself (called pardcap.c) with gcc using the following command:

```
gcc `pkg-config --cflags gtk+-3.0` -Wall -o pardcap pardcap.c `pkg-config --libs gtk+-3.0` -lm -lpng -ljpeg -lpulse-simple -lpulse
```

Once compiled, make the program executable using the chmod command:

```
chmod +x ./pardcap
```

You may then optionally put pardcap in a place that is on your PATH environment variable so you don’t need a local copy of it wherever you intend to run it.

The following assumes you have PARD Capture on your system ready to run. At the time of writing this manual only a Linux version of PARD Capture is available.

Starting the program: The main window

First ensure your camera is connected to the USB port.

Although PARD Capture can be started by ‘double-clicking’ the executable on many systems, it is preferable to start the software from a terminal window by typing ‘pardcap’ (or ‘./pardcap’ if it is not on your PATH) at the command prompt. This is because PARD Capture outputs information to stdout (to the console window) as it runs which gives useful updates and diagnostic data, warnings and error message details. All this information will not be visible to you if you start the program by simply ‘double-clicking’ it. The command line options are:

```
./pardcap [option] [argument]
```

The possibilities for [option] [argument] are currently:

```
-v
```

```
-h
```

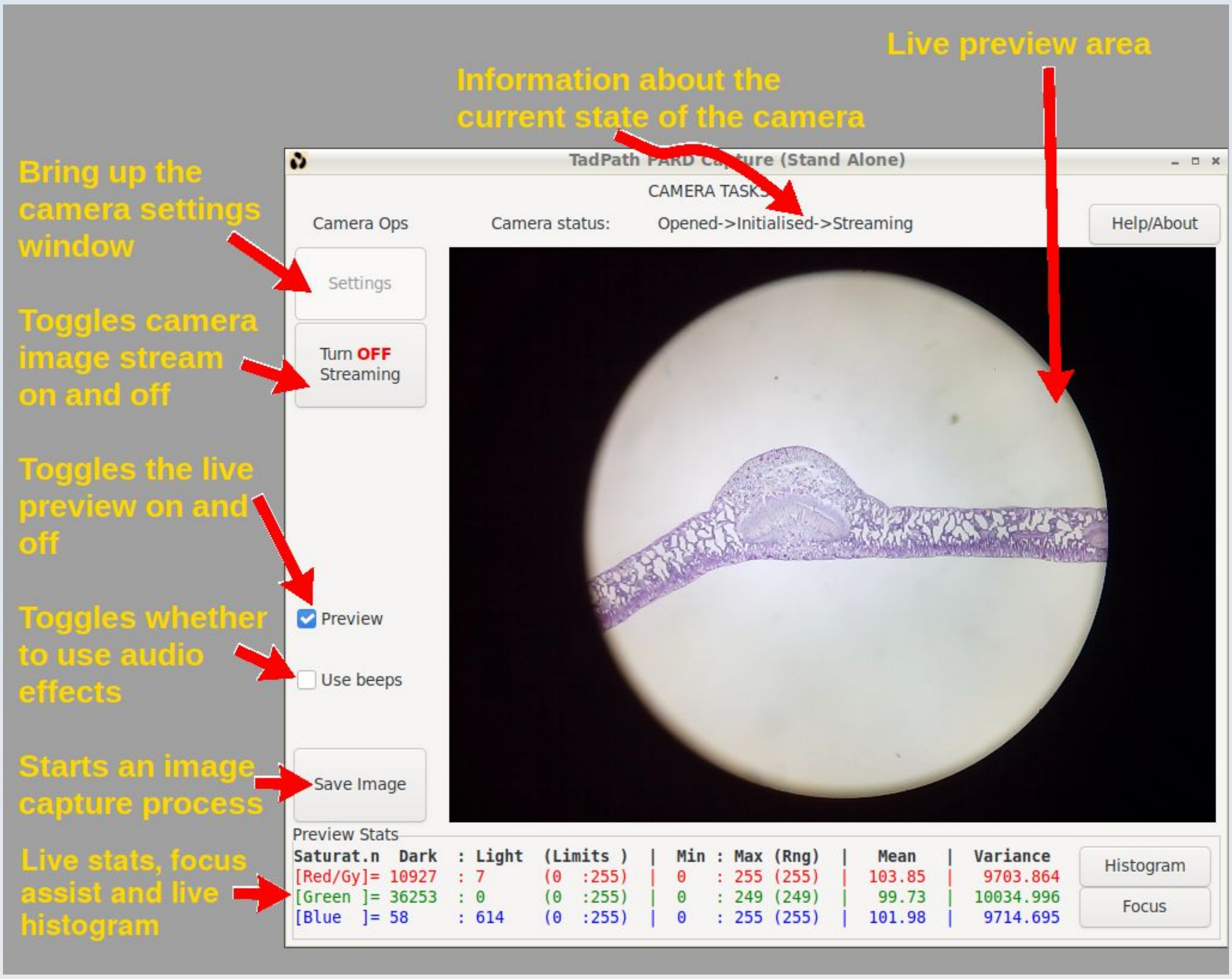
```
-l <logfile_name>
```

The -v option prints the program version number and must have no argument after it.

The -h option prints a help message and must have no argument after it.

The -l option must be followed by a file name (without spaces in the file name) which the user has permissions to create for writing and appending. This file will be created (so over-writing any existing file with that name) and all output messages will be recorded in this log file as a plain text file (so you may want to give it a .txt extension but that is not mandatory). If the user does not have access permissions to the chosen file then a pop up error message will alert the user that logging can’t be done but the program will then continue as usual. The start up screen (main window) is shown in figure 6.1.

The **camera status** line is a line of plane text above the preview window that tells you the current state of the camera. This may be opened or closed, initialised and streaming. Only when the camera is streaming can images be detected by the program for live preview and capture. Once a camera device is opened it remains opened (you cannot close it with this software unless you quit the program via the normal route by clicking on the ‘X’ of the main window and select to ‘Quit’ – only then will the camera device be closed by the software as part of its ‘Tidy up’ exit routine). Likewise once a camera is initialised it cannot be uninitialised manually via the GUI controls (although on occasion the software will uninitialise and re-initialise the camera internally as part of its various functions). The camera will be uninitialised when you quit the program by the normal route.



Live preview area

Information about the current state of the camera

Bring up the camera settings window

Toggles camera image stream on and off

Toggles the live preview on and off

Toggles whether to use audio effects

Starts an image capture process

Live stats, focus assist and live histogram

Camera status: Opened->Initialised->Streaming

```
Preview Stats
Saturat.n  Dark  : Light  (Limits )  | Min : Max (Rng)  |  Mean  | Variance
[Red/Gy]= 10927  : 7      (0  :255)  | 0   : 255 (255)  | 103.85 |  9703.864
[Green ]= 36253  : 0      (0  :255)  | 0   : 249 (249)  |  99.73 | 10034.996
[Blue  ]= 58     : 614    (0  :255)  | 0   : 255 (255)  | 101.98 |  9714.695
```

Figure 6.1 The main window and start up screen of PARD Capture. See text for further description.

The ‘**Settings**’ button will display all the settings window (to be described below) when clicked.

The ‘**Turn ON Streaming**’ button will do just that when clicked – it will attempt to start the camera stream. If successful then the camera status line will show that the camera is ‘streaming’ and also the button will change to show ‘**Turn OFF streaming**’ (as shown in figure 6.1) so if you click it in this state it will stop the camera stream of images. It is generally a good idea to turn off streaming when you are not using the camera for live previewing or for image capture because the camera will heat up more when it is streaming images.

The ‘**Preview**’ check box allows you to toggle on and off the live preview of images from the camera. This will only have any effect if the camera is streaming so if you see nothing when you check this box make sure the camera is streaming and turn on streaming if it is off (see the preceding paragraph). Regardless of the full image resolution of the camera image stream, the live preview will always be scaled to fit the

640x480 (VGA) sized live preview area. This will not affect the size of any captured images.

The ‘**Use beeps**’ check box enables you to get audio feedback from the program that may be helpful (e.g. if you are taking images in the dark). When you check this box to turn on beeps you will hear 3 high pitch beeps to acknowledge that you are now using beeps. When you un-check this box you will hear a single longer lower pitch beep to acknowledge that you ave turned off beeps. The other audio signals are as follows:

- The grabber took too long to get a frame (error signal that capture has been aborted): A quick decreasing scale of 5 beeps from high pitch to low pitch.
- Accumulating frame (for each frame of a multi-frame average): A single beep of medium high pitch.
- Start of a pre-capture countdown: Three beeps decreasing in pitch.
- Beginning of a capture event (individual, average or sequence): Two beeps of medium high pitch. This acknowledges that you activated the ‘Save Image’ button.
- End of a capture event: Three long low pitch beeps.

The ‘**Save Image**’ button is used to start a ***capture event***. A capture event involved saving one or more captured frames to disc. It is one of the following:

1. Take a single frame picture.
2. Take a multi-frame average picture.
3. Take a series of several pictures (each of which can be a single frame picture or a multi-frame average picture but which of these two must be constant for any given series).

Which of these 3 occur depends on how the user has configured the ‘Custom settings’ (see later). In any case, the capture event may start immediately or it may start after a time delay count-down, the duration of which is set via the settings controls discussed later. So in that case activating the ‘Save Image’ button will start the count down followed by the capture event.

When a series capture is in progress the ‘Save Image’ button will change its text to a bold red ‘Cancel Series’ label. Clicking the button when it shows this label will terminate the series capture sequence after the current picture is saved.

The **live stats area** will give a display of certain stats derived from the preview image (rather than the original size image from the camera). The full resolution of the preview image is used (no binning or downsampling is performed). Stats are calculated for each colour channel when the preview is in colour or for the monochrome signal otherwise.

The number of pixels (in the preview image) with intensity values below or equal to a lower limit and those with values greater than or equal to an upper limit (called **saturation limits**) are counted as 'saturation' pixels. Note that the intensity values read from a YUYV camera stream may never go below some finite positive limit (like 16) and never go above some upper limit that less than 255 (e.g. 235). It is for occasions such as that where the user's ability to edit the saturation limits (via the settings window) to make them different to the default (0 and 255) is particularly useful − otherwise you will never detect any saturation in these stats.

Stats are calculated from the full frame of the image in the preview window. This takes into account aspect ratio so the 'full frame' will be smaller than VGA if the original image aspect ratio is wider or narrower format compared to 4:3. Stats can also be restricted to only being calculated for pixels in the support of an optional preview mask (discussed in more detail later).

The **histogram** button toggles a live semi-transparent histogram display overlay on the preview. You can alter the position of this histogram on the display using the square bracket keys (see below) so it does not obstruct any preview image features of interest. When a colour preview is used you get 3 histograms overlain on each other – one for each colour channel.

Histograms are calculated on the range of 0-255 with a bin size of 1 grey level.

The user can select to have **cumulative histograms** plotted instead of the usual type via the settings window. This choice of usual vs. cumulative can be set independently for each colour channel. This choice affects the display of the histogram data but does not alter the histograms themselves (so does not affect any calculations done using the histogram data like mean and variance). See figure 6.7, right. The scale of the histograms can be automatic or manually set. See below for detail on these settings.

Mean and Variance are calculated from the histogram but the user can choose whether to have these calculations done using the whole histogram or only using the histogram values that are greater than the lower saturation limit and less than the upper saturation limit (said limits being set by the user in the settings window). This can be particularly useful when using variance as a focus assist measure.

The **focus button** allows you to toggle a **focus assist bar** onto the overlay. The focus assist facilities consist in the real-time update of certain image stats and the ability to do real-time Laplacian second derivative edge detection on the live preview stream. These stats may be simply observed as numbers in the live stats area below the preview window or they may be visualised in a graphic focus assist bar overlain on the top or bottom of the preview image (you can toggle the position using the ‘=’ key).

For colour previews each colour channel can be used alone for focus assistance or all together. For the graphic overlay focus bar you cycle between which colour channel value to use by clicking the 'Focus' button over and over − it will cycle between red, green, blue, all three (R,G,B) and off. For monochrome previews it will only cycle between on and off.

Key bindings: As an aid to working in dark conditions where moving a mouse about to specific locations may be impractical, and to get quick access to certain options, there are some keyboard strokes that will perform certain functions as long as the main program window has focus. They are:

- Press 'g' to activate the 'Save Image' button
- Press 'h' to print out a help message to stdout (the console window).
- Press 'a' to toggle the ‘Use beeps’ on/off.
- Press the 'c' key to close the camera connection
- Press the '[' or ']' keys to move the preview overlay
- Press the '=' key to toggle the focus bar to top/bottom of the overlay

The audio toggle (‘a’) is particularly useful in a headless situation (where you don’t have a monitor or the monitor is switched off to avoid light pollution) because you can use it to detect when the main program window has focus (and so when the other key bindings will work). If you press ‘a’ and don’t hear either of the ‘Use beeps’ toggle sounds then you know the main window is not in focus. Press Alt+Tab and try again. If still no sounds press Alt+Tab twice and try again. If still no sounds press Alt+Tab three times, etc. until you hear the beeps’ on/off toggle sounds, at which point you know that the main window has focus.

The 'c' option causes the camera connection to be closed. This is so you can re-establish connection when the camera connection fails without having to physically unplug and re-plug the USB cable or re-start the program and so loose any current setup. An 'Are you sure' popup precedes this action to avoid inadvertent closure of the camera connection. Presently this only safely allows for disconnection and re-connection of the same (model of) camera. If you try to connect a camera of different specs it may or may not work well and some labels in the settings window GUI may get messed up so if you want to change to a different camera the preferree method is to quit the program and re-start with the new camera connected.

The '[' and ']' keys are used to scan the histogram overlay across the image to find an optimum position. The move the histogram in steps in a raster scan and recycle at top/bottom.

The **preview area** is where the live preview image will be displayed when previewing is active. When live previewing is off the ‘Preview is OFF’ label will show, otherwise it will not be present. This is because when you toggle off previewing the last live image shown will remain in the display area so it may be confusing not knowing if what you are seeing is a live preview without any action going on or the still frame from the last previewing session. This label will make it clear which. One potential source of confusion here is that, when you are in live preview mode, it is possible to stop the stream from the camera and still be in live preview mode. In this case you will also see the last frame streamed from the camera as a still frame in the preview area but the ‘Preview id OFF’ label will not be showing (because preview mode is still active – i.e. the program is still looking for images coming from the camera, it is just that the camera is not currently serving up any images because the image stream has been turned off’. This situation can be clarified by simply looking at the camera status line above the preview area because this will show if the camera is streaming or not (and also to look at the ‘Turn ON/OFF Streaming’ button because it will show ‘Turn ON Streaming’ if the stream is currently off) – compare figures 6.1 and 6.2.

The preview area is a constant size being at VGA resolution (640x480) pixels and it cannot be scaled. When wide format images are being streamed they will show in this area at full width but with the top and bottom of the preview area greyed out. When a smaller resolution than VGA is being streamed then the image will be upsampled to match the VGA preview area size. When a larger than VGA image is being streamed then the higher resolution area will be downsampled to match the VGA preview size. All upsampling and downsampling preserves the original image stream aspect ratio.

For image streams of higher resolution there is an option in the settings window you can set called ‘**Use crop from full sized image as preview?**' With this option enabled then you will be able to zoom in to a 1:1 scale at any region of the preview by simply left-click (once) on the region you want to zoom in to. This mode will only activate for native camera stream resolutions that are greater than 640 pixels in width and/or greater than 480 pixels in height. For resolutions of VGA and below selecting this option will have no effect. Once you are zoomed in to a selected region you click again anywhere in the preview area to go back to the scaled down full preview mode.

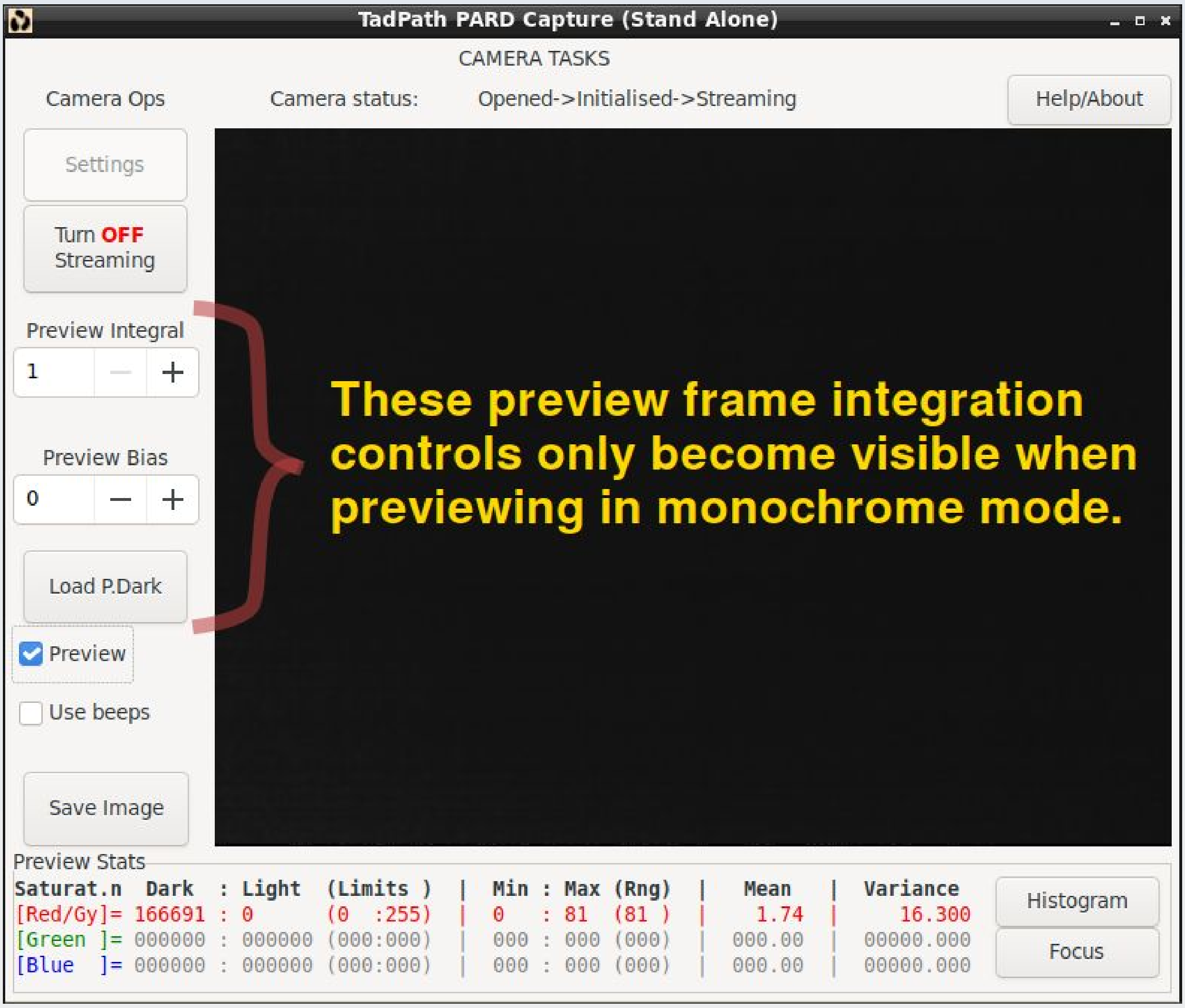
Preview frame integration (night vision)

There is a special variant of the live preview that allows for multi-frame integration for enhanced vision in dark conditions.

This frame integration mode is only activated with the ‘**Preview in monochrome?**’ options in the settings window is activated.

When this monochrome option is set the main window controls change to those shown in figure 6.2a

Note that this integration process only affects the preview. It has no impact on any actual captured image. Note also that this ‘integration’ mode and all its features and options are available and functional even if only 1 frame is ‘integrated’ (i.e. no actual multi-frame integration takes place).



Camera status: Opened->Initialised->Streaming

Preview Integral: 1

Preview Bias: 0

☑ Preview

□ Use beeps

These preview frame integration controls only become visible when previewing in monochrome mode.

Preview Stats

```
Saturat.n  Dark   : Light  (Limits )  |  Min : Max (Rng)  |   Mean   |  Variance
[Red/Gy]= 166691 : 0      (0  :255)  |  0   : 81  (81 )  |   1.74   |    16.300
[Green ]= 000000 : 000000 (000:000)  |  000 : 000 (000)  |  000.00  |  00000.000
[Blue  ]= 000000 : 000000 (000:000)  |  000 : 000 (000)  |  000.00  |  00000.000
```

Figure 6.2a Preview frame integration controls.

The ‘**Preview Integral**’ is a number from 1 to 20. The number shown here is the number of frames that will be integrated before the final resulting compound frame is displayed. A rolling integral procedure is used so as not to decrease the live frame rate. For example if you are integrating 5 frames then the process will start with just one frame being shown. Then when the next frame in the live preview sequence is captured it will be added to the first and the updated integrated 2 frame composite will be shown. Then a 3-frame composite, then a 4-frame composite, then a 5-frame composite. After that, the first frame will be discarded from the composite and the 6th captured frame will be added to the composite. So every new frame will be added to the composite and every oldest frame will be discarded from the composite. This has the effect of preserving the live preview frame rate at the expense of showing time-delayed trails of moving objects in the scene.

Note that this is integration, not averaging, so the final result will therefore get lighter overall (assuming each individual frame is not completely black with zero pixels all over – as may occur if you are using a preview master dark frame as described below). Because of this, lighter pixels in each individual frame can get saturated in the integrated result. To give you some control over this there is a ‘**Preview Bias**’ control. The number you select here is added to the integral result prior to display and so can shift the overall brightness of the resulting integrated frame. If you use a negative number here then very bright regions that would otherwise display as saturated white regions can be brought down back into the 0 – 255 greyscale display range to reduce or eliminate saturation. You can also use a positive number here to lighten the overall result. The available range for Preview Bias is from -4845 to +512.

Note that preview frame integration and bias calculations are all performed in integer arrays in the program (that is, the ‘int’ data type in the C programming language) so can handle very large resulting negative and positive intensity values (at least ± 32768 for computers with 2-byte integers but many modern computers have an ‘int’ that is 4 bytes long so the range is ± 2147483648). The resulting integrated and bias-shifted intensity values in the frame buffer are cropped into the range 0-255 for display only after all corrections have been applied.

Preview dark current correction and preview flat field correction multifunction button.

The button shown in figure 6.2a labelled as ‘**Load P.Dark**’ is actually a multi-function button used to load and eject optional master dark and master flat correction images for the live preview. This button can be cycled into three modes: ‘Load P. Dark’, ‘Load P.Flat’ and ‘Eject All’. The latter option (‘Eject All’) will not be available unless a custom preview master dark or master flat image is currently loaded. How to cycle the button will be described shortly, but first I'll give an explanation of these modes.

An internal preview master dark image and an internal preview master flat image is always present and actively applied when previewing in frame integration mode (even if only 1 frame is being 'integrated'). The internal master dark is always subtracted from each live preview frame first and then the resulting dark-frame corrected image is divided by the internal master flat. This correction is applied to each frame grabbed for preview and is done internally in a raw doubles buffer. This correction is applied to each frame prior to the frame being added to a multi-frame integrated result. When the program first starts the preview master dark image is set to all zeros so has no effect and all pixels in the preview master flat image are set 1.0 so this also has no effect. Note that these preview correction images (and any custom image you load to replace them as described below) only affect the live preview image. They have no effect on the captured image. For the captured image you can select separate (or the same if you chose) files for dark and flat correction as described elsewhere.

When the multi-function button is in 'Load P.Dark' mode and you click it, it will bring up a file chooser dialog to allow you to select a custom file to load and replace the existing internal preview master dark. According to the description just given above, if the file successfully loads, the effect will be immediate. If file load fails for any reason the existing preview master dark will be set back to its initial value (all pixels = 0.0) – i.e. it will be 'nullified'. When a custom preview master dark image is successfully loaded, the button font will change bold and colour that indicates which custom preview corrections files are loaded: Red means only a preview master dark is loaded but not a preview master flat. Green means that only a preview master flat is loaded but not a preview master dark. Blue means that both a master dark and a master flat preview correction image are loaded. A valid preview master dark must be in raw doubles format with .dou extension, a .qih external header file and it must be of the same dimensions as the preview image (currently this is VGA i.e. 640 pixels wide by 480 pixels high). You may a create a preview master dark in any way that generates a compatible raw doubles image file as defined above.

When the multi-function button is in 'Load P.Flat' mode and you click it, it will bring up a file chooser dialog to allow you to select a custom file to load and replace the existing internal preview master flat. If the file successfully loads, the effect will be immediate. If file load fails for any reason the existing preview master flat will be set back to its initial value (all pixels = 1.0) – i.e. it will be 'nullified'. Prior to use, the loading procedure will automatically 'normalise' the supplied master flat image i.e. it will find the mean pixel value in it and will divide all pixels in the loaded master flat by that mean to produce an image of corrective coefficients. This normalisation process will fail (resulting in failure of successful load procedure and nullification of the master flat) if the mean pixel value of the originally loaded master flat image is less than 0.5. The colour and weight of the font on the multifunction button will change upon successful load of a preview master flat as describe in the previous paragraph.

If you want to clear the currently loaded preview master dark and flat images then you must cycle the multifunction button till it is in ‘**Eject All**’ mode, then click it to perform the ejection (ejection nullifies both masters).

Cycling the multifunction button (figure 6.2b) is done by simply clicking it then either continuing with the operation or cancelling the operation. For example, if you click the multifunction button when it is in ‘Load P.Dark’ mode you have a choice of going ahead and loading a preview master dark image or you can click ‘Cancel’ on the file chooser. Whichever of these two options you follow, as soon as the file chooser is closed you will see that the button has moved on to the next function in its cycle (remember, ‘Eject All’ will not show unless a custom preview master dark or master flat is loaded). Similar behaviour occurs when you click the ‘Load P.Flat’ button. If you click the multifunction button when it displays ‘Eject All’ you will get a pop-up confirmation box that will give you the option to either go ahead with the ejection or cancel your request for ejection – either way when you respond to that box the function of the multifunction button will have cycled.

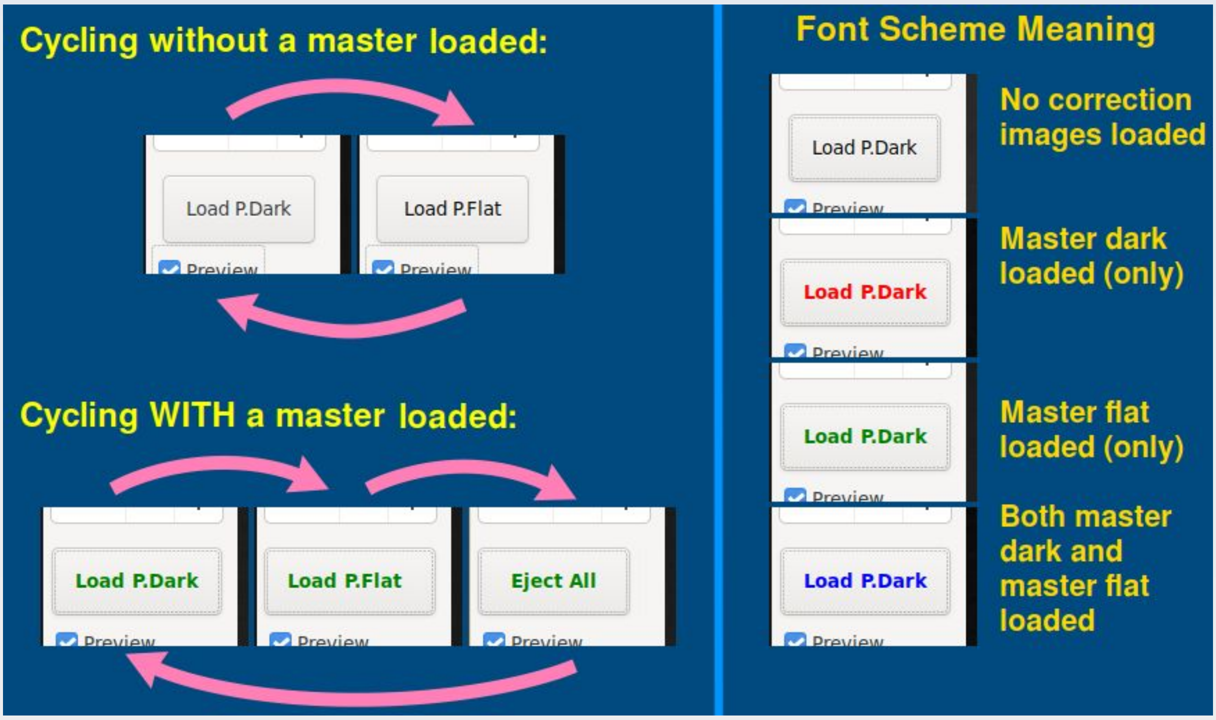

Cycling without a master loaded:
Load P.Dark
Load P.Flat
Preview
Cycling WITH a master loaded:
Load P.Dark
Load P.Flat
Eject All
Font Scheme Meaning
No correction images loaded
Master dark loaded (only)
Master flat loaded (only)
Both master dark and master flat loaded

Figure 6.2b The cycling options and font colour meaning on the preview master dark/flat field multifunction button. *Left*: the cycling behaviour depend on whether a custom preview master dark or master flat image is

loaded or not. *Right*: The button text font weight and colour indicate what custom correction files are loaded. This font scheme applies to any button text even though only the ‘Load P.Dark’ label is shown here.

The settings window

When you bring up the settings window by a click to the ‘Settings’ button in the main window) the program will query the camera firmware to find all the settings and resolutions and frame rates which the camera supports. It will then list those settings as GUI controls for you to adjust them. This means that if you run the program with a camera other than the AF51 you will see a different list of ‘User’ and ‘Camera’ controls (unless the camera just happens to have the same set of controls as the AF51). What follows is therefore specific to the settings available with the AF51 camera.

The settings in this window are divided into three groups (you will need to scroll to the bottom of the window to see them all – figures 6.3 and 6.4):

- User Controls
- Camera Controls
- Custom Controls

On the left hand side of the settings window are GUI selection controls where you may set your desired value for each of the settings. These selection controls are mostly in the form of a text entry box but sometimes a drop-down menu or a check box instead. For those cases where a file needs to be selected, a button is provided that brings up a file chooser dialog box so you can select a file name.

In the middle column of the settings window is a list of the currently applied value for each setting.

On the right is a description of the setting as well as an acceptable range statement.

At the top of the settings window are two buttons used to apply the selected settings: ‘**Apply ALL Settings**’ button which does what it says – it applies all the values that you have chosen via the GUI selection controls so they become the values that are currently applied and so replace the values shown in the middle column. This procedure includes applying all settings that require communication with the camera to change camera resolution and image stream format, exposure, colour balance and other settings.

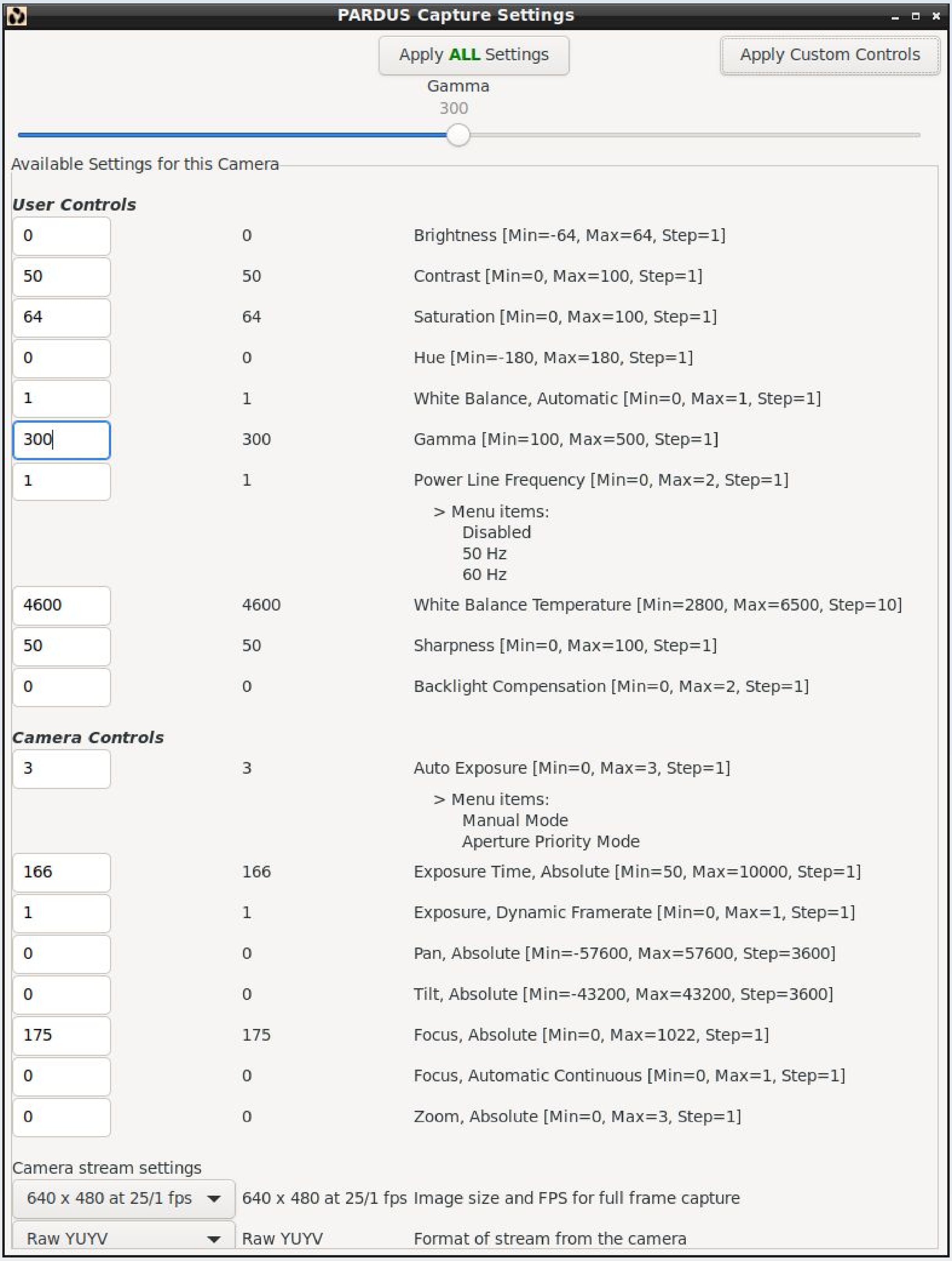
The other top button is labelled ‘**Apply Custom Controls**’. This only applies the settings that do not change any aspect of the camera hardware function like exposure, etc. The reason it may be useful to have this option is that when you access the camera hardware to change settings, this can sometimes cause some ‘one touch’ settings (like exposure) to revert to a default or automatic state and so this could mess up image capture sessions that depend on keeping all the camera settings constant between captures. You can therefore alter the file name of the captured image (for example) without disturbing the camera settings by using this button instead of the 'Apply ALL dettings' button.

Whatever button you use to apply your chosen settings, If you enter a value that cannot be set (e.g. a value that is out of the acceptable range) then an error will be presented to you and, if that value is one of the 'User' or 'Camera' control numerical values, the value that could not be set is highlighted in red in the middle column (this value in red is the value that remains in effect because it could not be changed to the one you supplied).

The 'User' and 'Camera' settings may be changed and applied in real time without having to click the 'Apply All Settings' button by means of a **real-time control slider** at the top of the window just under the 'Apply All Settings' button. To use this slider to change a setting value, left-click once in the text entry box for the setting you want to change. The name of that setting will appear above the slider to indicate that this is the control whose value the slider will be changing. The slider will automatically go to the position associated with the current value applied for that setting. This is illustrated for the 'Gamma' control in figure 6.3 You can then change the value of that control in real time by sliding the slider. This real-time control slider is not available for any of the 'Custom' settings.

The meaning and behaviour of all the 'User' and 'Camera' settings available for the AF51 have already been described in great detail in chapter 3 so will not be repeated here.

The '**Image size and FPS for full frame capture**' drop down list shows all the image stream sizes and maximum frame rate for each one that are available for the type of stream format that is currently selected. It is important to realise that if you select to change the '**Format of stream from the camera**' setting then the available image dimensions and maximum frame rates offered by your newly selected format may differ from the choices shown with the current 'image size and FPS for full frame capture' drop down list. For the AF51 camera all the image sizes available for the MJPEG image stream format will also be available for the YUYV format so image frame size will not be an issue when using this camera. However, the maximum FPS frame rate will differ between the two stream formats. For MJPEG 30 FPS will be permitted for all image sized but none of the YUYV stream frame sizes can be streamed at that rate so you will need to reduce the preview frame rate to a figure that is compatible with the chosen frame size resolution when changing from MJPEG to YUYV. Unfortunately, the list of maximum frame rates for the YUYV stream will not be shown in the GUI until you actually apply the changes. It is not a big problem however because all that will happen if you choose too high a preview frame rate is that you will get some warning pop up messages when you apply the changes and the program will automatically select the highest frame rate available for your chosen image size.



Gamma
300

Available Settings for this Camera

User Controls

0	0	Brightness [Min=-64, Max=64, Step=1]
50	50	Contrast [Min=0, Max=100, Step=1]
64	64	Saturation [Min=0, Max=100, Step=1]
0	0	Hue [Min=-180, Max=180, Step=1]
1	1	White Balance, Automatic [Min=0, Max=1, Step=1]
300	300	Gamma [Min=100, Max=500, Step=1]
1	1	Power Line Frequency [Min=0, Max=2, Step=1] > Menu items: Disabled 50 Hz 60 Hz
4600	4600	White Balance Temperature [Min=2800, Max=6500, Step=10]
50	50	Sharpness [Min=0, Max=100, Step=1]
0	0	Backlight Compensation [Min=0, Max=2, Step=1]

Camera Controls

3	3	Auto Exposure [Min=0, Max=3, Step=1] > Menu items: Manual Mode Aperture Priority Mode
166	166	Exposure Time, Absolute [Min=50, Max=10000, Step=1]
1	1	Exposure, Dynamic Framerate [Min=0, Max=1, Step=1]
0	0	Pan, Absolute [Min=-57600, Max=57600, Step=3600]
0	0	Tilt, Absolute [Min=-43200, Max=43200, Step=3600]
175	175	Focus, Absolute [Min=0, Max=1022, Step=1]
0	0	Focus, Automatic Continuous [Min=0, Max=1, Step=1]
0	0	Zoom, Absolute [Min=0, Max=3, Step=1]

Camera stream settings

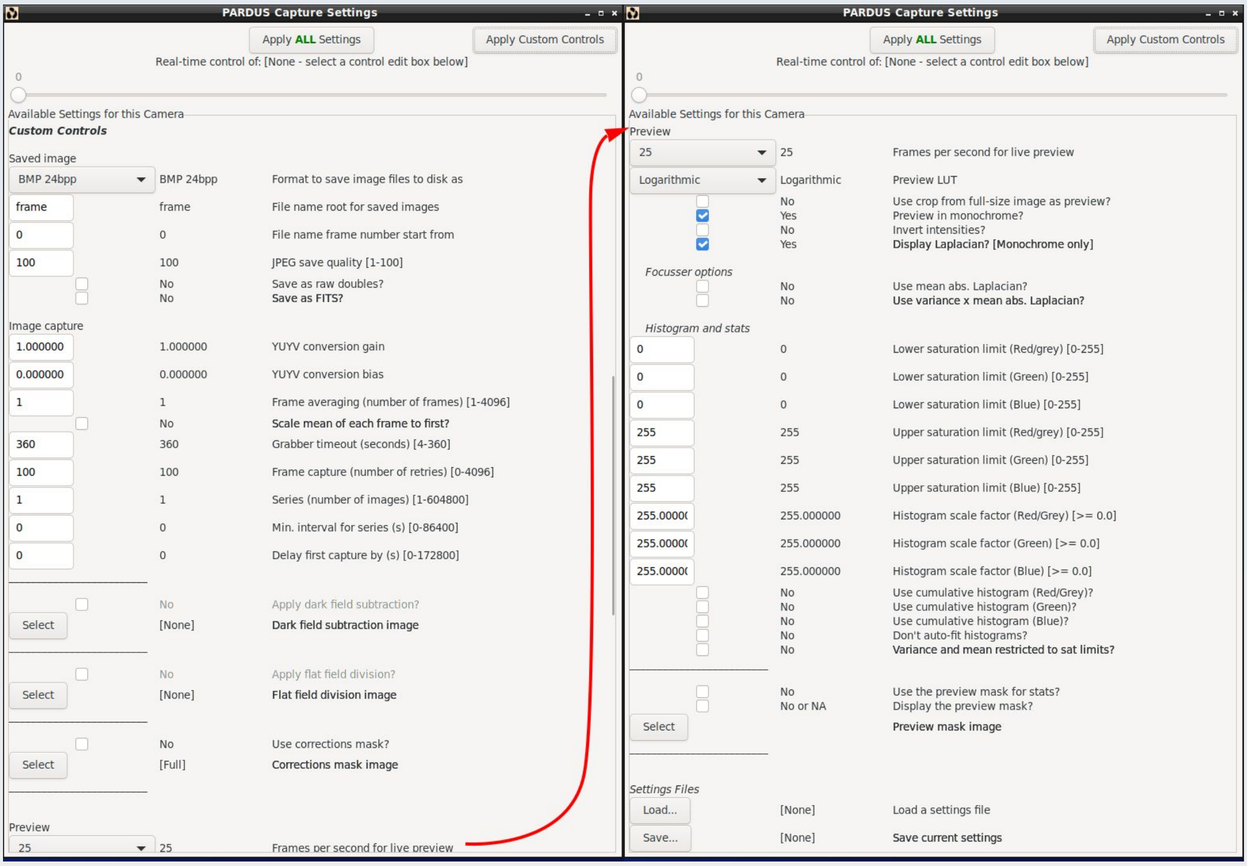
640 x 480 at 25/1 fps	640 x 480 at 25/1 fps	Image size and FPS for full frame capture
Raw YUYV	Raw YUYV	Format of stream from the camera

Figure 6.3 The ‘Settings’ window showing the ‘User’ and ‘Camera’ control settings available for the AF51 camera. These settings require communication with the camera hardware to change. This is done via the ‘Apply ALL Settings’ button or via the interactive slider but not the ‘Apply Custom Controls’ button. In this example the ‘Gamma’ value has been selected and so may be interactively varied by the slider at the top. If you select another value you will be able to control that value interactively with the slider at the top instead.

The ‘**Format stream from the camera**’ setting allows only one of two choices: MJPEG or YUYV, the latter being a raw uncompressed image stream from the camera. The AF51 supports both of these options. If you use a camera that only supports one or the other then you will not be able to change to the unsupported option.

Custom settings

All the ‘Custom’ settings are provided by the PARD Capture program itself and are not dependent on the camera firmware so they will be available for any camera that can use the PARD Capture software. These custom settings will be updated and applied when you click either of the two top ‘Apply’ buttons. Unlike the ‘Camera’ and ‘User’ controls, these ‘Custom’ controls can not be changed in real time with the slider control.



Real-time control of: [None - select a control edit box below]

Available Settings for this Camera

Custom Controls

Saved image

BMP 24bpp	BMP 24bpp	Format to save image files to disk as
frame	frame	File name root for saved images
0	0	File name frame number start from
100	100	JPEG save quality [1-100]
□	No	Save as raw doubles?
□	No	Save as FITS?

Image capture

1.000000	1.000000	YUYV conversion gain
0.000000	0.000000	YUYV conversion bias
1	1	Frame averaging (number of frames) [1-4096]
□	No	Scale mean of each frame to first?
360	360	Grabber timeout (seconds) [4-360]
100	100	Frame capture (number of retries) [0-4096]
1	1	Series (number of images) [1-604800]
0	0	Min. interval for series (s) [0-86400]
0	0	Delay first capture by (s) [0-172800]
□	No	Apply dark field subtraction?
Select	[None]	Dark field subtraction image
□	No	Apply flat field division?
Select	[None]	Flat field division image
□	No	Use corrections mask?
Select	[Full]	Corrections mask image

Preview

25	25	Frames per second for live preview
Logarithmic	Logarithmic	Preview LUT
□	No	Use crop from full-size image as preview?
☑	Yes	Preview in monochrome?
□	No	Invert intensities?
☑	Yes	Display Laplacian? [Monochrome only]

Focusser options

□	No	Use mean abs. Laplacian?
□	No	Use variance x mean abs. Laplacian?

Histogram and stats

0	0	Lower saturation limit (Red/grey) [0-255]
0	0	Lower saturation limit (Green) [0-255]
0	0	Lower saturation limit (Blue) [0-255]
255	255	Upper saturation limit (Red/grey) [0-255]
255	255	Upper saturation limit (Green) [0-255]
255	255	Upper saturation limit (Blue) [0-255]
255.0000(	255.000000	Histogram scale factor (Red/Grey) [>= 0.0]
255.0000(	255.000000	Histogram scale factor (Green) [>= 0.0]
255.0000(	255.000000	Histogram scale factor (Blue) [>= 0.0]
□	No	Use cumulative histogram (Red/Grey)?
□	No	Use cumulative histogram (Green)?
□	No	Use cumulative histogram (Blue)?
□	No	Don't auto-fit histograms?
□	No	Variance and mean restricted to sat limits?
□	No	Use the preview mask for stats?
□	No or NA	Display the preview mask?
Select		Preview mask image

Settings Files

Load...	[None]	Load a settings file
Save...	[None]	Save current settings

Figure 6.4 Scrolling down the settings window we have the ‘Custom’ controls. These are the same for any camera that can be used with the PARD Capture software. See text for details.

The ‘**Format to save the image files to disk as**’ is the format any saved images will be saved as. The ‘save-as’ options and their meanings are as follows:

Raw YUYV – This is the YUYV frame direct from the camera. The saved file will have an ending of the form '_yuyv.raw'. This will be a raw array of unsigned integers at 16 bits per pixel (bpp), i.e. unsigned shorts. There will be 1 integer for every pixel so the array will be if total size: height x width integers. The integers will be arranged in raster fashion from top left (pixel [0]) to bottom right (pixel [(height x width) - 1]). Each 16 bit integer will contain two 8 bit values: the first byte is a luminance value and the second byte is a colour value (the first colour value from a pair of pixels is Cb1 from the first pixel and Cr1 from the second pixel). This means that a full colour value will only be available for half the total number of pixels, each full colour value pair (the Cb1,Cr1 pair) being applied to the two contiguous pixel positions from which their components were derived. There will be no header information saved (neither an internal header nor an external header filed). Because this just dumps the raw data off the camera it requires no colour conversion pre-process so it is the quickest method to grab an image from the YUYV stream but for the same reason it has the following restrictions. This save-as format will not be allowed when the camera stream format is MJPEG. If you select this option when the camera is strsaming in MJPEG format and try to save an image you will get a pop up error telling you it cannot be saved. You will not be allowed to do multi-frame averaging, masking, dark field correction or flat field correction when using this raw YUYV format. The options to save a raw doubles or FITS version of the file (to be discussed later) will be ignored – no such files will be saved even if you check the box to save them.

Y PGM – This saves only intensity data as a grey level image in PGM p5 format. The saved file will have an ending of the form '_Y.pgm'. If the camera is streaming in YUYV format the data will be just the extracted Y components of the stream. If the camera is streaming in MJPEG format the data will be the I (intensity) component of an RGB-to-HSI transform calculated as I=(R+G+B)/3.

Y BMP – This saves only intensity data as a grey level image in 8 bpp BMP format. The saved file will have an ending of the form '_Y.bmp'. If the camera is streaming in YUYV format the data will be just the extracted Y components of the stream. If the camera is streaming in MJPEG format the data will be the I (intensity) component of an RGB-to-HSI transform calculated as I=(R+G+B)/3.

Intensity – This saves only intensity data as a grey level image in raw doubles format (i.e. double precision floating point binary) with an external BiaQIm header file (i.e. a .qih file). The saved file will have an ending of the form '_I.dou' and the external header will be a plain text file with the same name as the image but with the extension .qih instead of .dou. If the camera is streaming in YUYV format a full YUYV to RGB conversion will take place for each frame. If the camera is streaming in MJPEG format a full RGB decoding will always be done. In either case (YUYV or MJPEG) the data saved will be the I (intensity) component of an RGB-to-HSI transform calculated as I=(R+G+B)/3. So the 'Intensity' save-as format differs from the Y PGM and Y BMP in that it will always use the full colour information to derive intensity even if the camera is streaming in YUYV format (and it will always save the data in raw doubles format). Also, because it uses the full RGB data up until the end, if you want to do dark field subtraction or flat field division corrections (see below) you must supply a 3-channel (colour) master dark image or master flat image (see figure 6.5).

PPM – This saves a 24 bpp colour image in the PPM P5 format. The saved file will have an ending of the form '_rgb.ppm'. If the camera is streaming in YUYV format a full YUYV to RGB conversion will take place for each frame. If the camera is streaming in MJPEG format a full RGB decoding will always be done.

BMP 24bpp – This saves a 24 bpp colour image in 24 bpp BMP format. The saved file will have an ending of the form '_rgb.bmp'. If the camera is streaming in YUYV format a full YUYV to RGB conversion will take place for each frame. If the camera is streaming in MJPEG format a full RGB decoding will always be done.

PNG – This saves a 24 bpp colour image in 24 bpp PNG format (no transparency). The saved file will have an ending of the form '.png'. If the camera is streaming in YUYV format a full YUYV to RGB conversion will take place for each frame. If the camera is streaming in MJPEG format a full RGB decoding will always be done.

JPEG – This saves a 24 bpp colour image in JPEG format. The saved file will have an ending of the form '.jpg'. If the camera is streaming in YUYV format a full YUYV to RGB conversion will take place for each frame. The image will be saved using libjpeg RGB to JPEG encoding regardless of whether multi-frame averaging is being done or not. If the camera is streaming in MJPEG format and multi-frame averaging is being done then a full RGB decoding will be done for each frame. The image will be saved using libjpeg RGB to JPEG encoding. All cases where libjpeg encoding is used, the encoding process with use a compression factor supplied by the user via the '**JPEG save quality [1-100]**' setting (1 being most compression and worst image quality and 100 being least compression and best image qualiy).

If the camera is streaming in MJPEG format and NO multi-frame averaging is being done then the frame grabbed from the MJPEG stream will be saved directly as a JPEG image (libjpeg will not be used and the user-supplied JPEG save quality value will be ignored). In this case also there will be no ability to save a raw doubles or FITS image, and no masking, dark field or flat field correction regardless of whether the user has specified those processes or not. This is a very quick and direct way to just grab a JPEG still from the camera stream.

The '**File name root for saved images**' is a string that is used to construct the file name for the saved images. It should not contain any full-stops, spaces or other special characters that might cause issues with file names. The program does not check for these things so it is up to you to ensure the root string is compatible with your system. This forms the first part of the file name and will not change from capture to capture. The complete file name will be constructed as described below.

The '**File name frame number start from**' is an integer that will form part of the file name for a saved image. This integer will increment each time an image is saved and will only reset when the user clicks the 'Apply All Settings' button (when it will set to whatever the user has put in this entry box).

The complete file name used to save an image is constructed in the form:

```
root_####ending
```

where ‘root’ is the string supplied for the ‘**File name root for saved images**’ setting, ‘####’ is an integer that starts with the value supplied for the ‘**File name frame number start from**’ setting and ‘ending’ is the ending described under each format of the ‘**Format to save the image files to disk as**’ setting described above.

The ‘**Frame averaging (number of frames)**’ is an integer which specified how many frames are to be captured and averaged together to form the output saved image. The default when you start the program is 1 meaning that no multi-frame averaging will be done. During a multi-frame average capture the current frame being accumulated into the average is shown on stdout (the console) and you will hear a beep for each frame as it is accumulated if you have set ‘Use beeps’. Some averages of many frames, especially high resolution frames in dark conditions, may take a long time to complete. While the program is performing an average you will see a new button appear on the main window labelled ‘**Cancel Averaging**’ (figure 6.5, left). If you click this button the current multi-frame averaging sequence will be terminated and whatever frames were accumulated up to that point will be output as an average of that many frames. During a series capture (described below) if you press the ‘Cancel Averaging’ button it will only cancel the current multi-frame averaging sequence, it will NOT cancel the series. The partial average will be saved and the next image in the series will commence its capture.

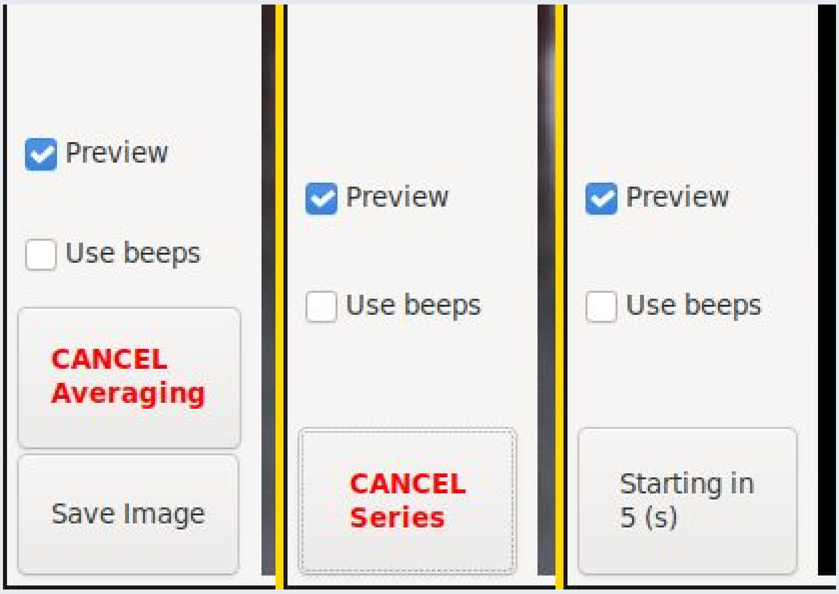


Figure 6.5 Variations in the buttons on the lower left aspect of the main window which show up under various circumstances. The cancel buttons that appear in the main window when in multi-frame averaging and/or series capture mode. *Left*: Cancel averaging button. *Middle*: Cancel series button. Both these buttons will appear simultaneously if you are capturing a series of multi-frame averages. **You may need to wait for the effect of pressing either cancel button to take place because the program only checks the GUI events infrequently during an image series or during a multi-frame average sequence. So click once and leave the mouse pointer over the button till the click has been registered (the button will change appearance).** *Right*: A live countdown timer replaces the ‘Save Image’ button label when you click it if the ‘Delay first capture by (s)’ setting is more than 0. In this case you can abort the count down (and the capture event) by left clicking (once) on this button while the count down is in progress.

All multi-frame averaging is done in double precision floating point accumulator buffers. Once all the frames have been accumulated the result is divided by the denominator (the number of frames accumulated) using a double precision floating point denominator. The end result is clipped into the 0-255 range for 8 bpp and 24 bpp outputs but it left in its double precision floating point form for output as raw doubles and FITS.

The corrections mask support (see below) is not used for multi-frame averaging – full support is used always even if a custom corrections mask has been supplied.

When the program interfaces with the camera stream it has to request a frame from the camera and then wait for the camera to respond. Sometimes, especially with the higher resolutions in raw YUYV streaming format, it can take the camera a long time to respond. The two settings '**Grabber timeout (seconds)**' and '**Frame capture (number of retries)**' allow you to set how patient the program is in waiting and retrying if the camera is taking too long to serve up an image. If these values are exceeded without an image being retrieved from the stream then the program will give up trying and pop up an error message. All capture events will be terminated (whether a single frame capture, a multi-frame average or a series capture). So, if you are doing a long series capture or multi frame averaging – especially with low light level conditions and high resolution frames it will be best to keep these value high to avoid your process being prematurely terminated.

The '**Series (number of images)**' value is an integer which tells the program to capture a series of images at set time intervals, automatically, after you first click the 'Save Image' button. The default value is 1 meaning that series capture mode is inactive. The '**Min. interval for series (s)**' lets you set the time interval between captures in a series. It therefore has no effect if you are not in series capture mode. Using these settings allows you to capture a time lapse sequence of images. Each image in the series may be a multi-frame average according to your chosen value for 'Frame averaging (number of frames)'. The interval is called the 'Min.' (minimum) interval because the actual interval will be determined by how long the program needs to wait to capture each image in the series. This may be significantly more than 1 second if you are capturing high resolution images in the dark and if you are using multi-frame averaging for each image.

When capturing a series, a log file is automatically generated in the current working directory called 'Series_root.txt' where 'root' will be replaced by whatever string you entered for the 'File name root for saved images' setting. This log will show the date and time of the beginning and end of the capture series, the names of the images captured and the actual interval (in seconds) between each frame (the time that elapsed since the last frame was captured to the time the current frame is captured) and the reason for termination of the series.

When capturing a series the label on the 'Save Image' button will change to '**Cancel Series**' (figure 6.4, right). Pressing this during a series capture will have the effect of cancelling the current multi-frame average sequence (if multi-frame averaging for each image in the series is being done) similar to clicking the 'Cancel Averaging' button AND ALSO it will terminate the series such that after the current image has completed being saved, no more images in the series will be started and the 'Save Image' button will return to its normal appearance.

The '**JPEG save quality**' setting has been described above in the 'JPEG section of 'Format to save the image files to disk as'.

The ‘**YUYV conversion gain**’ and ‘**YUYV conversion bias**’ settings are floating point numbers which represent the gain and bias used in converting the raw YUYV values from the camera’s YUYV stream into RGB values for further processing in the program.

The ‘**Delay first capture by (s)**’ setting allows you to set a count-down timer for delayed activation of the ‘Save Image’ button action. When you click the ‘Save Image’ button, this number of seconds will pass before the effect takes place – whether that is to capture a single frame, a multi-frame average or a series. The countdown delay only occurs for the first activation after you click the ‘Save Image’ button so, in the case of a series, there is no countdown delay for subsequent images in the series and there is no countdown delay for every frame in a multi-frame average. During the delay the button label will change to a live count down timer (see figure 6.5, right) and if you left click the button while this count down timer is in progress you will abort the count down as well as the image capture event. This facility is provided because if you are going to do a long capture series or multi-frame average then you may want some time to get away from the scope before the capture starts to avoid any vibrations from you moving around or out of the room being picked up in the capture. As another example you can set the delay for several days if you want the capture to start at a quiet period when no one is around and there is minimal traffic (e.g. on a weekend night in a lab) again to minimise contamination by vibrations. Another example would be that you want to capture in the dark so you need some time to switch off the monitor and other lights after you click ‘Save Image’.

The ‘**Save as raw doubles?**’ option, when selected, will save any image due to be saved as raw doubles in addition to whatever save-as format you selected with the drop down list described above. This option will be ignored if the save-as format selected from the drop-down list is ‘Raw YUYV’ or if it is ‘Intensity’ (because that will be saved as raw doubles anyway) or if it is ‘JPEG’ ***and no*** multi-frame averaging is being done. If it is JPEG and multi-frame avaraging ***is*** being done then this option will be honoured.

The naming convention for these raw doubles files will depend on the save-as format. For monochrome formats the extension is changed to ‘.dou’. For colour formats, 3 separate files are output with the ‘ending’ is changed to ‘_R.dou’, ‘_G.dou’ and ‘_B.dou’ for each primary colour channel. In all cases each file will be accompanied by an external BiaQIm header file with the same name as the raw doubles image file but with the extension ‘.qih’.

This option to save as raw doubles is provided so you can take advantage of the increase in quantisation resolution afforded by multi-frame averaging. This is also the only format you can use to load in an image for use as a master dark field image for dark field subtractive correction and one of the formats you can use to load a master flat field image for use in divisional flat field correction (see below).

The ‘**Save as FITS?**’ option, when selected, will save any image due to be saved as a FITS file in addition to whatever save-as format you selected with the drop down list described above. The FITS file will include additional information about the frame capture and some settings in its header and the pixel data will be saved as binary double precision floating point values. This option will be ignored if the save-as format selected from the drop-down list is ‘Raw YUYV’ or if it is ‘JPEG’ ***and no*** multi-frame averaging is being done. If it is JPEG and multi-frame averaging ***is*** being done then this option will be honoured.

The naming convention for these FITS files will depend on the save-as format. For monochrome formats the extension is changed to '.fit'. For colour formats, 3 separate files are output with the 'ending' is changed to '_R.fit', '_G.fit' and '_B.fit' for each primary colour channel.

This option to save as FITS with double precision binary float values is provided so you can take advantage of the increase in quantisation resolution afforded by multi-frame averaging. Currently FITS files can only be saved, they cannot be read by PARD Capture as input for dark field and flat field corrections (for those input files you can use the raw doubles option described above). This FITS output option is provided for those who want to capture images with PARD Capture and then do further processing in other software that can read FITS files. The FITS files produced by PARD Capture are compliant with the FITS standard.

Optional pre-processes

These settings relate to optional pre-processes that may be applied to frames as they are being captured and before they are output.

All these pre-processes are performed in double-precision floating point internal frame buffers so are not restricted by the 8 bpp or 24 bpp ranges of the original captured images.

All these pre-processes are only applied within the support of the current corrections mask (see below) so that only pixels under the non-zero pixels in that mask will be used in the calculations and corrections.

Figure 6.6 shows the algorithm used to apply the optional pre-processes.

The '**Scale mean of each frame to first?**' option, when selected, is a pre-process applied to multi-frame averaging sequences. It therefore has no effect if multi-frame averaging is not being done. If this option is selected then, when a multi-frame average is first started, the first image frame that is captured has its mean pixel value evaluated and that mean value is stored. All subsequent frames in the multi-frame average also have their mean pixel value calculated and compared to the stored value of the first frame. Prior to being accumulated into the averaging buffer, the current frame is scaled by a constant multiplicative factor (applied to each pixel in the frame) that transforms its original mean to be equal to the mean of the first frame. The multiplicative scale factor used is calculated as:

```
scale_factor = Mean_of_first_frame / Mean_of_current_frame
```

Thus this option scales the mean 'brightness' of each image to be equal to the mean brightness of the the first image in the averaging sequence and so compensates for any fluctuations in illumination that my occur over the multi-frame averaging period. This ensures that no one frame dominates the resulting average result.

The feature may be particularly useful for the purpose of capturing a multi-frame average to be used as a 'master flat' image for flat field correction (see below). Note that during this process, all frame averages are calculated only from pixels that are in the support of the current corrections mask (see below) and only pixels in the support of the corrections mask have the scale factor applied to them. Furthermore, to avoid ‘divide-by-zero’ errors and floating point overflow errors, an arbitrary lower limit is placed on all frame means: if any frame has a mean pixel value that is less than 1.0e-10 it will be scaled by 1.0 and a warning message will be printed to stdout (the console) if that happens.

The ‘**Apply dark field subtraction?**’ option, when selected, subtracts the loaded master dark image from each frame captured. See the corrections algorithm in figure 6.5 for detail. This option cannot operate unless a suitable master dark image is loaded. ‘Suitable’ means the master dark selected by the user must have the same dimensions as the current image being captured and must have the same number of colour channels i.e. a monochrome single channel master dark image if you are capturing monochrome images and a 3-channel master dark image if you are capturing colour images. You chose your what master dark image to load with the ‘**Select**’ button below this check box. This brings up a file chooser dialogue box from which you can select a file. Only files in raw doubles format (.dou with a .qih external header file) can be used as input. In the case of loading a master dark field image for colour image correction, the file you chose must be one of a triplet of files that have the same prefix name but end in ‘_R.dou’, ‘_G.dou’ and ‘_B.dou’ (or the same but with lower case ‘r’, ‘g’, ‘b’). Each raw double image must have a corresponding external header file ending in ‘.qih’. You can load / choose any one of these three and it will be accepted provided the other two members of the group exist and can be read and are of the same dimensions to the one you chose. In the case of loading a monochrome master dark field image the file name must end in ‘_Y.dou’ or ‘_I.dou’ (or ‘_y.dou’ or ‘_i.dou’) and have a corresponding ‘.qih’ external header file. Such images with compatible file names for either colour or monochrome use can be generated by PARD Capture using the ‘Save as raw doubles?’ option or using the ‘Intensity’ save-as format described above.

The ‘**Apply flat field division?**’ option, when selected, divides each frame captured by a pre-processed master flat field image. See the corrections algorithm in figure 6.5 for detail. This option cannot operate unless a suitable master flat image is loaded. ‘Suitable’ means the master flat selected by the user must have the same dimensions as the current image being captured and must have the same number of colour channels i.e. a monochrome single channel master flat image if you are capturing monochrome images and a colour or 3-channel master flat image if you are capturing colour images. You chose your what master flat image to load with the ‘**Select**’ button below this check box. This brings up a file chooser dialogue box from which you can select a file. File formats that are acceptable for use are those with the following extensions: ‘.ppm’ (a PPM P6 formatted image), ‘,pgm’ (a PGM P5 formatted image), ‘.bmp’ (an MS Windows BMP formatted image of either 8 bpp or 24 bpp) and ‘.dou’ (a raw doubles image with an accompanying ‘.qih’ external header file). When using raw doubles images for colour image correction, the same rules apply as for loading a colour master dark field image (see above).

I mentioned that the master flat image is pre-processed prior to use for divisional correction. What this means is that the image you select and load, automatically undergoes the following processing prior to being finally accepted for use in flat field correction (if it fails this processing you will get an error message telling you that the file cannot be used for flat field processing). In the following description, all processing is only done for pixels within the support of the currently active corrections mask:

1. Find the maximum pixel value – if it is less than 0.5 the loaded image is not suitable for use as a master flat and so it is rejected with an error message presented to the user. (No realistic flat field correction image taken with a camera down a microscope should have a maximum pixel value of <1 assuming grey level readout is on an integer scale starting at 0.) This condition also prevents the possibility of divide-by-zero in step 3 below.
2. Find the mean pixel value.
3. Now normalise the image by dividing all pixel values by the mean.

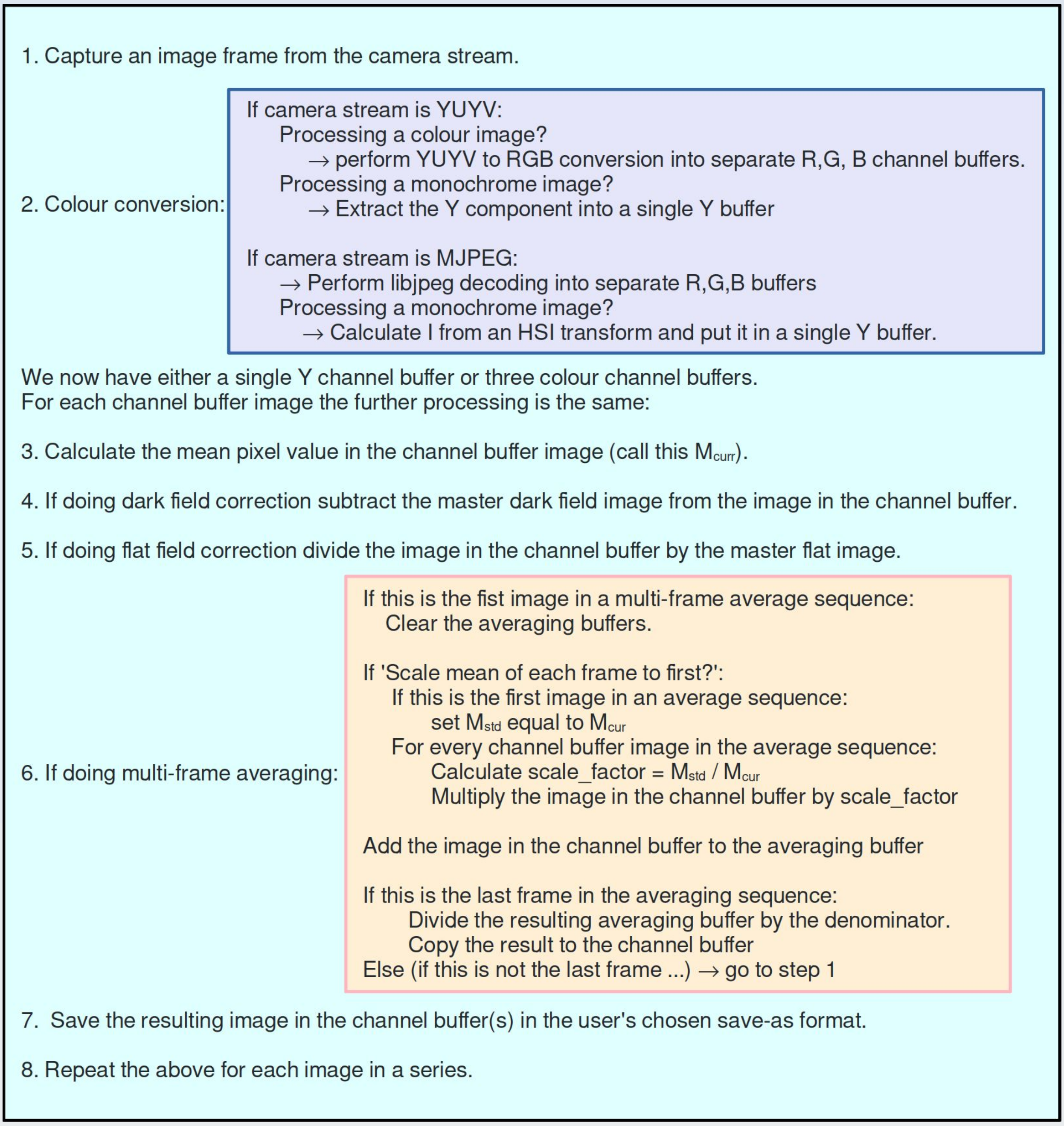
The master flat is now pre-processed and ready to use for flat field divisional correction. See below for the algorithm.

The '**Use corrections mask?**' option, when selected, uses a user-supplied custom binary mask image to mask all optional pre-processes. If no user-supplied custom mask image is provided or if this option is not selected then all optional pre-processes will be performed on all pixels of the frame (i.e. a full support mask will be internally generated).

You chose your what custom corrections mask image to load with the '**Select**' button below this check box. This brings up a file chooser dialogue box from which you can select a file. File formats that are acceptable for use are those with the following extensions: '.ppm' (a PPM P6 formatted image), ',pgm' (a PGM P5 formatted image), '.bmp' (an MS Windows BMP formatted image of either 8 bpp or 24 bpp).

In order to be accepted as a user-supplied custom mask, the custom chosen file must also pass the following tests and pre-processing (these take into account and operate on all pixels in the chosen mask image i.e. the mask image itself is not subject to masking):

The mask image must be the same height and width (in pixels) as the current main image. If it is not then this chosen mask is rejected with an error message being shown to the user and the default internal full-support mask is used instead (of course the user has the option to try again and load a different mask).



```
1. Capture an image frame from the camera stream.

2. Colour conversion:
    If camera stream is YUYV:
        Processing a colour image?
            → perform YUYV to RGB conversion into separate R,G, B channel buffers.
        Processing a monochrome image?
            → Extract the Y component into a single Y buffer

    If camera stream is MJPEG:
        → Perform libjpeg decoding into separate R,G,B buffers
        Processing a monochrome image?
            → Calculate I from an HSI transform and put it in a single Y buffer.

We now have either a single Y channel buffer or three colour channel buffers.
For each channel buffer image the further processing is the same:

3. Calculate the mean pixel value in the channel buffer image (call this $M_{curr}$).

4. If doing dark field correction subtract the master dark field image from the image in the channel buffer.

5. If doing flat field correction divide the image in the channel buffer by the master flat image.

6. If doing multi-frame averaging:
    If this is the fist image in a multi-frame average sequence:
        Clear the averaging buffers.

    If 'Scale mean of each frame to first?':
        If this is the first image in an average sequence:
            set $M_{std}$ equal to $M_{cur}$
        For every channel buffer image in the average sequence:
            Calculate scale_factor = $M_{std}$ / $M_{cur}$
            Multiply the image in the channel buffer by scale_factor

    Add the image in the channel buffer to the averaging buffer

    If this is the last frame in the averaging sequence:
        Divide the resulting averaging buffer by the denominator.
        Copy the result to the channel buffer
    Else (if this is not the last frame ...) → go to step 1

7.  Save the resulting image in the channel buffer(s) in the user's chosen save-as format.

8. Repeat the above for each image in a series.
```

Figure 6.6 Algorithm for the optional pre-processes. Note that for a ‘save-as’ format of ‘Intensity’ this is treated as a colour image all the way to the end and only at the saving step is the intensity component calculated from the resulting R,G,B channel buffers and saved to disc. This means that if you want to do dark field or flat field correction with a ‘save-as’ format of ‘Intensity’ then you must supply an RGB format image as the master dark or master flat image even though the output will be monochrome.

Regardless of image save-as type (whether colour or monochrome) only one identical mask is applied to all colour channels. For this reason, if a colour format mask image was chosen the data in the three colour channels is reduced to a single channel by extracting intensity component of an HSI and using that for the following processes. The formula is I=(R+G+B)/3.

The supplied mask pixel values are thresholded about 127 to force them to have values of either 0 or 255 thus: if (pixel_value >127) pixel_value = 255 else pixel_value = 0

The resulting mask image is then tested for support – it must have at least 1 pixel of support (i.e. an all zero mask is not allowed). If not then the mask is rejected with an error message being shown to the user and the default internal full-support mask is used instead (of course the user has the option to try again and load a different mask).

If the mask passes all the above it is used for masking the optional pre-processes ONLY when the 'Use corrections mask?' box it ticked, otherwise the internal full support mask is used instead.

Live preview options

The '**Frames per second for live preview**' setting means the frame rate of the live preview and does not affect the frame rate of the image stream from the camera. However, you will not be allowed to have a preview FPS that is greater than the maximum frame rate of the image stream from the camera. If this preview frame rate is higher than the maximum image stream rate you will get the warning messages referred to above when you apply the settings. Bear in mind that this is a maximum possible frame rate – the actual number of frames per second of the live preview will depend on the actual receive frame rate from the camera and the depends on other factors such a the exposure setting, the speed and processing power of the host computer and the current speed of the USB connection to the camera.

Another important issue to bear in mind is that, especially for some of the higher resolution image streams from the camera, performing some of the post processing options discussed below like Laplacian stats for focus assistance and some LUT transforms may cause the GUI to partially freeze until the frame rate is reduced to a level your computer can cope with. If this **freezing of the GUI** occurs you will notice that the GUI buttons do not change their appearance when you click on them check boxes will not change from checked to unchecked and drop down combo boxes will not drop down. Also the live preview will not update despite being on. If this happens you can recover by doing one of these things:

1. Reduce the live preview FPS and click 'Apply' buttons at the top of the settings file (even if they appear unresponsive).

2. If you can't lower the FPS then 'uncheck' whichever pre-processes you have set (in particular the Laplacian options discussed below) and click one of the 'Apply' buttons. Remember, you may not see a change in the visible 'checked' status of the check box you click on – don't worry about that, just click it once if it appears checked and click 'Apply'

3. Try switching off live preview by clicking on the preview on/off check box in the main window.

The ‘**Preview LUT**’ combo box offers a list of look up table (LUT) transforms to apply to the live preview to make certain features stand out as you require it. These transforms do not alter the actual pixel values of the preview and so do not alter the stats calculated from them including any used for focus assistance. The default is ‘Linear’ which is a 1-to-1 mapping of the image grey levels with intensities. Other options are ‘Inverted’, ‘Logarithmic’, ‘Exponential’, ‘Square_root’ and ‘Square’. The inverted option can be particularly useful if you chose to invert the actual preview image intensity values for the purpose of helping with focus assistance (see below) because this ‘Inverted’ LUT will -re-invert’ the preview image so it looks normal (non-inverted). The ‘Logarithmic’ options is particularly useful when viewing the Laplacian preview image (see below).

The ‘**Use crop from full-size image as preview?**’ option, when selected, will allow you to zoom in on an region of the preview image so you can see the preview in 1:1 pixel scale for those image sizes that are larger than VGA in any dimension. It has no effect for camera image sizes that are VGA or smaller. When this mode is active the live preview window will show a central label ‘Click to zoom’ which invites you to select the region you want to zoom into. When you cleft-click (once) with your mouse over a part of the prevew, that part will expand to full 1:1 pixel resolution of the native camera stream size format and the central ‘Click to zoom’ label will disappear. Left click again (once) anywhere in the preview window to re-scale the preview image to wholly fit into the preview window and so have the option to select another region to zoom into.

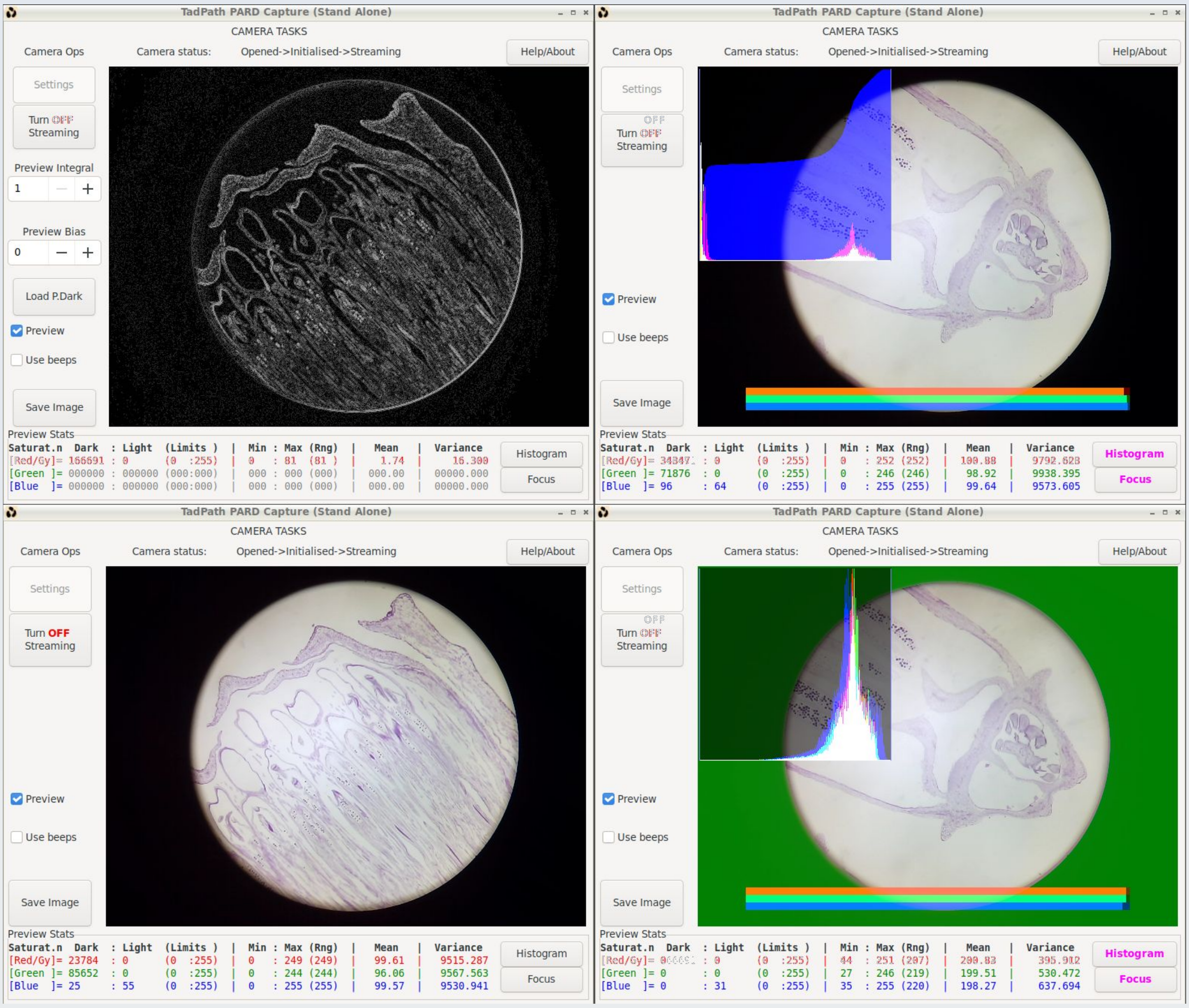
The ‘**Preview in monochrome?**’ option, when selected, shows the live preview only in greyscale, discarding all colour. For a camera stream format of YUYV this saves time because only the Y values are extracted directly and no colour conversion need be done. If the camera stream is in MJPEG mode this is actually *more* work for the computer because MJPEG stream images need to be converted to full RGB anyway so you will be adding the extra step of finding the intensity from the RGB data using the formula I=(R+G+B)/3. However, in either case, a particular advantage of selecting this monochrome option is that it gives you the ability to do real-time rolling multi-frame integration for enhanced dark vision preview monitoring. See the section on ‘Preview frame integration (night vision)’ for more detail. Also, only in monochrome preview mode can you see a live Laplacian magnitude gradient image.

The ‘**Invert intensities**’ option changes all the pixel values to their inverse by the transformation:

```
new_value = 255 – original_value
```

This is done for each colour channel in a colour preview and for the intensity values in a monochrome preview. The result is a ‘negative’ image. The pixel values are alters prior to any stats (including the Laplacian measures described below) being calculated on the preview image. Note that this will not affect the Laplacian measures because only absolute values of the Laplacian are calculated. If you chose this invert option to help with focus assistance but don’t want to look at a negative image you can restore the non-inverted image appearance using the ‘Inverted’ LUT option described above.

The ‘**Display Laplacian?**’ option is only effective when previewing in monochrome mode. This converts the intensities in the preview into the absolute second partial derivative to give a kind of ‘edge’ image (figure 6.7 top left). This is done prior to histogram data collection so it affects the values derived from the histogram – the range, saturation pixels, mean and variance. The Laplacian image is usually quite dark so may be better visualised with a logarithmic LUT.



```
Saturat.n  Dark   : Light  (Limits )  |  Min : Max (Rng)  |   Mean   |  Variance
[Red/Gy]= 166691 : 0      (0  :255)  |  0   : 81  (81 )  |    1.74  |    16.300
[Green ]= 000000 : 000000 (000:000)  |  000 : 000 (000)  |  000.00  | 00000.000
[Blue  ]= 000000 : 000000 (000:000)  |  000 : 000 (000)  |  000.00  | 00000.000
```

```
Saturat.n  Dark   : Light  (Limits )  |  Min : Max (Rng)  |   Mean   |  Variance
[Red/Gy]= 34847  : 0      (0  :255)  |  0   : 252 (252)  |  100.88  |  9792.628
[Green ]= 71876  : 0      (0  :255)  |  0   : 246 (246)  |   98.92  |  9938.395
[Blue  ]= 96     : 64     (0  :255)  |  0   : 255 (255)  |   99.64  |  9573.605
```

```
Saturat.n  Dark   : Light  (Limits )  |  Min : Max (Rng)  |   Mean   |  Variance
[Red/Gy]= 23784  : 0      (0  :255)  |  0   : 249 (249)  |   99.61  |  9515.287
[Green ]= 85652  : 0      (0  :255)  |  0   : 244 (244)  |   96.06  |  9567.563
[Blue  ]= 25     : 55     (0  :255)  |  0   : 255 (255)  |   99.57  |  9530.941
```

```
Saturat.n  Dark   : Light  (Limits )  |  Min : Max (Rng)  |   Mean   |  Variance
[Red/Gy]= [illegible] : 0  (0  :255)  |  44  : 251 (207)  |  200.83  |   395.912
[Green ]= 0      : 0      (0  :255)  |  27  : 246 (219)  |  199.51  |   530.472
[Blue  ]= 0      : 31     (0  :255)  |  35  : 255 (220)  |  198.27  |   637.694
```

Figure 6.7 Some preview options. Top left: Live absolute Laplacian image in monochrome preview mode seen with a logarithmic LUT. The original colour image of this same specimen is seen bottom left for comparison. Top right: Live histogram and focus assist bars overlain on the preview. In this example the blue channel histogram is shown as a cumulative histogram and the red and green channel histograms are displayed as usual histograms. Note the extensive saturation on black / dark pixels because of the central round field of view down the microscope leaves a lot of peripheral pixels dark and not useful for calculating focussing stats. Bottom right: To remedy this situation a custom mask is used which excludes the peripheral pixels from calculations. Now the histograms (all channels shown as normal histograms, not cumulative) show almost no saturation.

The ‘**Focusser options**’ let you choose what stats to use as focus assist tools. The default focus assist measure is the variance. However you can change this to the mean absolute Laplacian based on a 3x3 Laplace kernel if you enable Laplacian pre-processing (check the '**Use mean abs. Laplacian?**' box). This uses the mean absolute Laplacian on its own. If you instead (or also) tick the '**Use variance x mean abs. Laplacian?**' then a compound statistic is used which, as the name suggests, is the variance multiplied by the mean absolute Laplacian. This can give a very sensitive measure of focus – even moreso if you increase the sharpening factor of the camera and your image has sharp edge features in it. The variance is the variance of the intensities of the displayed preview image (ignoring any LUT effects) which will be the intensity (not Laplacian) images for the individual colour channels of a colour preview and it will be either the intensity or the Laplacian image for the monochrome preview depending on whether the user has chosen to display the Laplacian for the monochrome preview or not as per the settings options.

While viewing the monochrome preview as a Laplacian image, the value for the image 'mean' of the image will be calculated from this Laplacian display image but will have a lower value compared to the value called '**Mean abs. Laplacian**' shown in the live stats display when the ‘Use mean abs. Laplacian’ check box is ticked, even though on first thought you might think these should be the same. The reason for the difference is that the when the Laplacian display image is constructed the result of the convolution at each kernel position is done in doubles and then converted to unsigned char to give the pixel value for the display. A one pixel boundary all around this display image is then set to zero to compensate for the fact that the 3x3 kernel can't go all the way to the boundaries of the image. The 'mean' of this image is calculated using those unsigned char values and the zero pixel boundary pixels are included in the denominator. For the ‘Mean Abs. Laplacian’ statistic used as a focus assist measure, this is constructed from the sum of the doubles result at each kernel position (without rounding to an unsigned char integer) and only pixels where the rounded unsigned char value is non-zero are used for this statistic. The denominator used to turn this sum into a mean only includes those non-zero pixels as well and excludes the zero boundary pixels. Due to this smaller denominator and the fact that fractional results are not rounded down in the numerator, this value will be larger than the 'mean' of the preview display Laplacian image.

When not using any of the Laplacian focus measures the mean and variance displayed in the real time stats will be those of the intensity images for each channel.

When the 'Use Mean abs. Laplacian?' box is ticked the mean value shown will be the mean absolute Laplacian calculated for focus assistance but the variance displayed will be the variance of the displayed preview image. When the 'Use variance x mean abs. Laplacian?' box is ticked the variance value displayed will be the variance multiplied by the mean absolute Laplacian. When both boxes are ticked the 'variance x mean abs. Laplacian' will be used for focus assist but the mean value displayed will be the mean absolute Laplacian. The value shown will be indicated by an altered header on the stats region of the preview window and the value used for focus assist will be highlighted in magenta so there is no need to memorise these rules – the visual cues will make it clear what value is being displayed and what value is being used for the focus assist bar(s). [There is no magenta highlight when only the default variance measure is used for focussing].

Using the focus assist bar

The **focus assist bar** continually (frame to frame) calculates the minimum value (of whatever focus measure you have chosen) encountered and the maximum value encountered for all frames displayed until you reset the session (see below for how a preview focus session is reset). During this session the current value of the focus measure is plotted on its position within the range defined by the current maximum and minimum. When you first start a session this range value will be very small (or actually zero for the very first frame of a session) so the current focus value position will appear to fluctuate wildly between extremes until you extend the range by changing the focus a little.

If, at any time, an invalid focus measure is encountered such as a value that is zero or very close to zero, then the focus bar will light up as yellow (regardless of what colour channel the bar pertains to) to let you know that the focus value is currently invalid. A brief flash or yellow may be seen whenever to start or refresh a focus session for the reason given above.

For colour previewing the focus statistics values will only update on the stats display when a focus bar overlay is active – and then only the values of the colour channel used for the focussing bar will be updated (all three will be updated if you chose to use all three channels for the focus bar – as shown in figure 6.7, right).

You can also modify the focus assist sensitivity by altering the number of frame integrals and bias values for the monochrome preview.

The focus assist bar overlay shows how close the current value of the focus assist measure is to the maximum it has been – the closer the bar is to the right of the overlay the closer the current preview image is to the one that gave the maximum value of the focus measure being used. Of course this 'maximum' value will change when you alter the image in some way, especially if you move the field of view or change contrast, sharpness, exposure, brightness, gamma, etc. So you can reset the focus session (to re-establish a new range including a new maximum value) in any of these ways:

1. Clicking or toggling the 'Focus' button on the stats part of the main window

2. Clicking either of the 'Apply' settings buttons on the settings window (even if you have not changed any settings).

3. Clicking on the preview window when you are in full-size tile mode (any time you click to zoom in or out the focus range resets)

The preview bar can use any or all or the red, green or blue channels of a colour image – just keep pressing the ‘Focus’ button to cycle through the options.

You can move the focus bar between top and bottom positions with the equals key (=).

Histogram options

You have a significant amount of control over how the live preview histograms and stats are scaled and calculated. Use the ‘**Saturation limit**’ boxes to provide your own limits above and below which pixels should be counted as saturation pixels. You will not be allowed to set the upper and lower limits equal for any channel or to have the upper limit lower than the lower limit value. The level of the current saturation limits will be marked as vertical lines overlaid on the histogram display – one pair of lines for each colour channel's limits.

All histograms are displayed in an area of 256 x 256 pixels overlaid on the preview image. The position of this area can be moved as described previously but its size never changes. By default the scale of each histogram is set automatically and independently for each colour channel by measuring the greatest bin value (call this Hgm_max) then scaling the histogram so that the whole range of histogram values fits into the range 0 – 255 pixels high. This scale is modified by whatever value you put in the '**Histogram scale factor**' boxes (call this sf). The default value for the scale factor is 255 so if you leave everything as default then each histogram will automatically fit into the height range of 0 to 255. If you alter the scale factor value so that the calculated height of any bin goes above 255, the value is just clipped to be 255. The calculation is:

```
Position for current histogram bin of value x =
                                   clip_to_within_0_255(sf * x/Hgm_max)
```

This situation is OK for some purposes but if there is a great deal of saturation the rest of the histogram will appear very small. You can overcome this by masking (described below). However, for colour images, as each colour channel histogram is scaled independently you will not get a sense of the quantitative relationship between the channels unless you fix the scales manually and avoid this automatic scaling calculation. For this reason you have the option to check the '**Don't auto-fit histograms?**' box. With this ticked, '`Hgm_max`' in the above formula will be replaced by a constant 1.0 for each colour channel. This means the quantitative relationship between the colour channels will be preserved but it also means that the default scale factor of 255 will usually be too high and the histograms will be mostly off the scale of the display area. So if you check this box you would normally also reduce the scale factors for each channel to about 0.05 (instead of the default 255). You can set a different scale for each channel but these scale factors will be constant so will not affect change of the histograms for each channel relative to each other.

The '**Variance and mean restricted to set limits?**' check box will make it so that the mean and variance calculations will only use pixels whose values are greater than the lower saturation limit and less than the upper saturation limit.

Using a custom preview mask

The default position is that all pixels in the preview window that are used for the image are also used to form the histogram and all stats derived from the histogram including saturation pixel counts and also any focus measure (including Laplacian-based measures). The only pixels that are excluded by default are those that do not form part of the image due to the image from the camera having a different aspect ratio to the 4:3 aspect of the display area.

However, there are many occasions when it may be useful to restrict stats and focus measure calculation to certain regions of the image only. A good example is the case shown in these figures where the field of view of the camera is round resulting in a periphery of unused images in the image that dilute any statics of image feature variation and make focussing measures less sensitive. Even in full-frame images without a black periphery, you may only want to focus on a small area with features of interest.

For such situations, you may load a custom '**Preview mask image**'. Using the 'Select' button to select a file. This will check the file is of a suitable format and size, namely 8 bpp grey scale or 24 bpp colour BMP file, or a binary pgm or ppm file and VGA size i.e. 640 pixels wide by 480 pixels high. If you select a satisfactory file, the file name will be kept and the file will only be read when you next press one of the 'Apply' buttons – but only if you also select to use this file via one of the associated check boxes, otherwise the file will not be loaded.

The mask file must be VGA size regardless of whether the original image stream from the camera is 4:3 or 16:9 or other wide or narrow format. For image streams that are wider than 4:3 the unused top and bottom rows of any supplied mask will just be ignored. Likewise any side pixels will be ignored for image streams that are narrower than 4:3.

When you chosen mask file is loaded it will be used to create a binary internal preview mask by simple thresholding. For mask files of grey scale format, the internal mask will be considered 'on' (stats will use that pixel position) if the mask file grey level is greater than 127 (so 128 and above). Any pixel in the mask file loaded from disc that has a value of 127 or below will translate into a zero pixel in the internal mask and that pixel position will not be used for calculating any stats. If the mask file loaded is a colour image then the intensity value will be calculated from the RGB values and intensity > 127.0 will result in the internal mask pixel being 'on' (all other pixels will be set to 'off').

You may visualise the mask as an overlay to the preview image using the check box '**Display the preview mask?**'. This only affects whether you see the mask on the preview or not – it has no relevance to whether the mask is used to mask off pixels for stats calculations. When visualised, the colour of the mask pixels that are zero will display with altered colours and the pixels where the mask is non-zero ('on') with display with unaltered colours.

If you want to actually use the mask to restrict what pixels get used for stats calculations you do that by using the separate '**Use the preview mask for stats?**' check box. Both these check boxes will be unusable (insensitive) until you select a valid image to use for preview masking via the preview mask file chooser dialog box (i.e. using the 'Select' button).

The ability to display a mask overlay without it affecting stats calculations can be used to allow for some custom overlay on the preview such as a counting grid, a scale bar or some other annotation of your choice which you supply as the custom mask image.

Saving and loading settings files

At the bottom of the settings window are two buttons '**Load ...**' and '**Save ...**' which bring up file chooser dialogue boxes that allow you to load or save a settings file. A settings file is a plain text file in a particular format that can be written and read by PARD Capture. Settings file names must have a .txt extension.

When you load a settings file, the settings will be checked to see if they are compatible with the camera currently attached to the system and that the file is of the correct format. Successful loading of a settings file will set values in the GUI entry controls to the values contained in the settings file as if you had entered those valued manually yourself. However the settings file load process will not apply the settings. The settings will be applied when you manually click the ‘Apply All Settings’ button at the top of the settings window. This gives you an opportunity to inspect the loaded values and modify them if you wish.

When you save a settings file, the file will contain all the currently applied control values and file names that are in effect (e.g. the file name of any master dark image, master flat field image or mask). You can use this facility to keep a record of a particular experimental set up and, by loading that settings file you can reproduce that setup for another session.

Default settings

Unlike some other camera software, PARD Capture does not have a ‘reset to defaults’ button or menu option which allows a user to change all the camera settings back to their default values (i.e. the settings values the camera has when it is powered off then on again).

However, this is a useful feature to have and PARD Capture can achieve the same functionality using settings files. To do this, when you first plug a camera into a USB port and start up PARD Capture go to the settings window and save a settings file before you make any changes to camera settings. Call this file something like ‘Defaults_Cam####.txt’ where ‘####’ is replaced by something that identifies that camera.

Now, any time you want to revert that camera back to its default settings simply load that defaults settings file and apply the settings.

Incompatibility between cameras

Because the range of available settings may be different for each camera a settings file saved when using one camera will probably not be readable when a different camera is connected to the system. It may be by chance that a different camera has the same list of named settings in its firmware to another camera so the settings files saved with one camera can be loaded with the other camera attached but the ranges and effects of those settings may differ between the two cameras. For this reason it is best to name the file according to the camera used in that session when you save a settings file (or place that settings file in a folder related to that camera or use some other way to identify which camera was used when a particular settings file was generated e.g. by adding comments to the settings file as described in the following section).

Rules for manually editing a settings file

This section may be skipped if you have no intention of manually editing a settings file.

The settings files are generated automatically by PARD Capture and must be of a particular format to be readable by it but a settings file may be manually edited ***provided*** you adhere to the rules and restrictions on editing described here. **Failure to adhere to these rules and restrictions could make the file unreadable by the program or worse – crash the program or spoil your camera**.

For this reason it is best not to manually edit these settings files other than to insert some comment lines for your own reference and documentation. If you feel you must edit it then only change setting values without altering any formatting and follow all the rules described below. If you have no intention of manually editing the file then there is no need to learn the rules.

Comments can be inserted on any line and must start with a hash (#). Comments may also come after ALL data on a line and must again start with a #. For example, the very first line of an example settings file might be

```
PCamSet 1.1 20 139
```

but it could be modified by you to be:

```
 PCamSet 1.1 20 139 # Any comment you like here ...
```

Bear in mind that the read buffer within PARD Capture for any line (whether a comment line or a data line with or without a following comment) is limited to MAX_CMDLEN-1 characters (this is currently 511 characters). If you have a single line that contains more than this number of characters the program will not function properly and may crash. Any leading space on a line is ignored for interpretation purposes but still counts towards the maximum line buffer length mentioned above. So if you want to make the file more 'human readable' by adding some custom indents to it, that is OK. Any trailing white space is also ignored for interpretation but also counts towards the maximum line buffer length.

The file has two sections: Settings read from the camera firmware and custom settings provided by the PARD Capture software.

The first line in the file must begin with PCamSet followed by the version number of the settings file (1.1 in the above example) followed by the number of settings retrieved from the camera itself (20 in the above example) followed by the number of widgets used in the settings window (139 in the above example). No blank lines or comment lines can precede this – it must be the very first line in the file. This marks the beginning of a settings file and it also marks the beginning of the section in the settings file that lists the settings from the camera firmware with each of those settings arranged in a block of lines.

You could re-arrange the order of these settings blocks but you **must** keep each data block together. For example you could mix up the order of the camera firmware settings blocks or you could even mix in some of the customs settings up at the beginning of the file. The file reader in the PARD Capture software does not care about that.

The camera firmware settings come in blocks. Each block begins with 'idx:' followed by an integer and the line after that is the name of the control (preceded by 'name:'. If you make any edits you ***must*** ensure that 'idx:' always comes immediately before its corresponding 'name:' line. Any number of blank or comment lines can come in between them but nothing else (the same is true for the other mandatory conditions described below).

If you make any edits you ***must*** ensure that the 'ctrl:' line comes before the 'curr:' line and the 'mdx:' line is the last line in a block except for any menu item names that come after it (you ***must*** ensure any menu item names that follow the 'mdx:' line are preserved). You ***must*** ensure all the other members of a block are kept together but the order in which they appear does not matter provided the rules are adhered to. For example 'curr:' could be moved to come before 'max:' if you like.

The custom settings come after the line that begins 'PCustSet' (the version number of the customs settings format appears after this). The version number must be correct and compatible with the version of the PARD Capture program for this file to be readable. The foregoing describes the position of the PCustSet line in a file that has been generated by PARD Capture. The actual position of the PCustSet line in this file is irrelevant to the PARD Capture program provided it is found before the end of the settings and provided it does not come before the PCamSet line.

Regarding the custom settings, these are all on one line for each setting and that line begins with a 'windex' label which must not be alered. The format of the information that follows each windex must not be altered – you can edit values but not the format. The order in which each windex line appears is not important.

A comment line is autogenerated by PARD Capture for each setting. Like any comment, these can be edited or removed at will without adverse effect.

The end of the file (as far as PARD Capture is concerned) is marked by the line that has the single word 'exit' on it. After this line PARD Capture stops reading the file and closes it. You may therefore write whatever free text you like after the 'exit' line in whatever format you like – those lines do not need to begin with a a comment hash character. Also the MAX_CMDLEN-1 characters limit no longer applies.

7. Achieving the best quality images with the AF51 – avoiding artefacts

The AF51 can be used in a simple ‘point-and-shoot’ mode but this will often not yield the best quality results or results suitable for scientific quantitative processing and analysis. This is because the image may be subject to artefacts with certain control settings and under certain conditions. Also the CMOS sensor will have some imperfections such as hot pixels and local variations in amplifier characteristics that only show up in very low light level conditions and need to be corrected for to get properly calibrated quantitative information from the images. This chapter will describe those artefacts and give pointers as to how to avoid or minimise them.

Uneven illumination – flat field correction

This can occur if the camera is not properly fitted to the eyepiece – see figures 5.6 and 5.7. Following the advice in chapter 5 will minimise this. In some circumstances and with some illumination systems on some microscopes you will not be able to completely flatten out the illumination across the field using the correct camera fitting procedure but only minimise it – the residual uneven illumination may still be significant for your purposes. Also, even if the illumination appears flat and good enough for making illustration pictures, there will nearly always be some subtle variations in sensitivity across the field due to dust particles on lenses, etc. which need to be corrected if you are doing various kinds of quantitative photometric image processing and image comparisons

Under these circumstances a master flat field image (a ‘***master flat***’) should be prepared and applied as flat field correction to the main image frames when you capture them. The PARD Capture software provides the facilities to both make a master flat field image and apply flat field correction during image capture. If you are using other software to capture your images from the camera you may have to perform flat field correction using multiple software packages over several imaging sessions to achieve the same effect.

When preparing a master flat field image a multi-frame average is advised to minimise noise in your master flat because any residual random noise pattern in the master flat will be transmitted to your main images that you will be correcting with it. In general you should first acquire a master dark field image (see below) then subtract this for every frame in your master flat average as you acquire it (using the ‘Apply dark field subtraction?’ option). You should also enable the ‘Scale mean of each frame to first?’ feature and, optionally, supply a corrections mask if there is dead space in the image (such as when imaging the circular field of view down an eyepiece). An example of a ‘master flat’ produced in this way using the AF51 on a PUMA microscope is shown in figure 7.1

I advise that you capture your master flat with a degree of illumination that will be typical for the actual picture image capture session. So if you are going to capture an scene in full bright field then make your master flat with full bright field illumination but if you are going to be capturing a low light level scene (e.g. fluorescence) then use a low light level for your master flat. Also use the same constant manual exposure setting (and other camera settings) for taking the master flat series as you will be using for the main image capture.

This is because variations in AF51 exposure setting will also impact on the gain setting and this in turn may alter the variation in light sensitivity across the field due to non-linearity of the sensor gain characteristics.

I also advise you make use of the ‘Save as raw doubles?’ option to capture all the enhanced radiometric information afforded by the multi-frame averaging procedure.

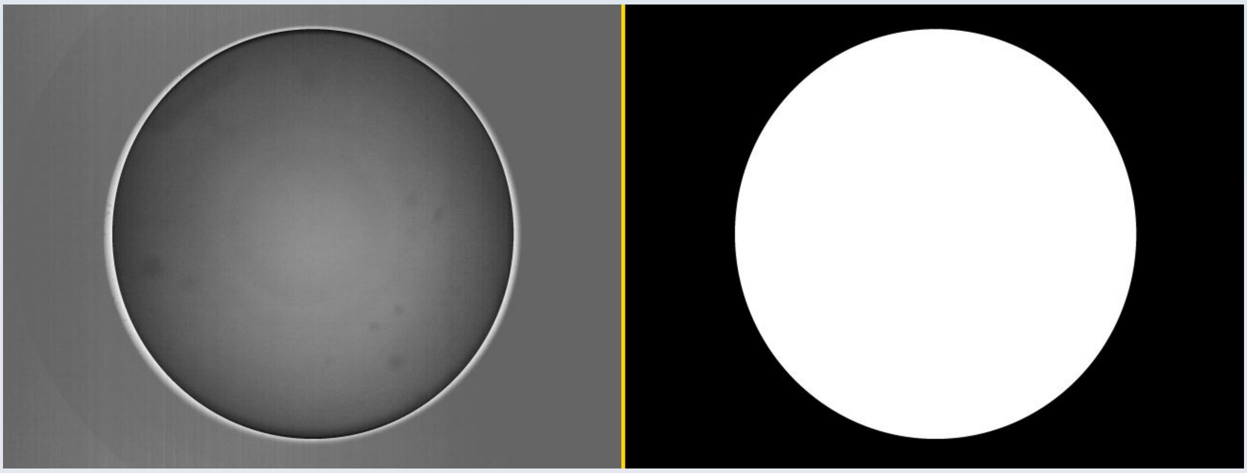


Figure 7.1 *Left*: A master flat field image made using the AF51 with PARD Capture. This was done using the Y-only signal from the camera and multi-frame averaging (only 64 frames were used in this example). The actual image is darker than shown here because I was making this to do low light level epi-fluorescence imaging but the grey levels have been stretched (contrast enhanced) for the purposes of this illustration to show you the blurry dust spots and field curvature which this master flat will be correcting for. *Right*: The corrections mask used when making this master flat. This accounts for the artificial sharp cut-off seen in the master flat image. I use a corrections mask in this case because the periphery of the camera field is not contributing to the imaging of the region of interest (the central round eyepiece field of view).

Quantisation artefacts and random noise

The raw output of each frame from the AF51 camera can distinguish 256 brightness levels per channel from 0 to 255 (maximum). This 8-bit depth of intensity quantisation causes ‘chunky’ steps in colour or grey levels in regions where the intensity varies only slightly. This is quantisation artefact – the actual intensity varies smoothly in real life but can only be digitised into discrete steps.

Such ‘chunky steps’ in variation of grey scale or colour will also be made worse by the use of compressed video streams such as MJPEG or compressed image save-as formats such as JPEG. This form of chunky or blocky image regions will affect spatial resolution variation and not only intensity variation. These are compression artefacts.

Examples of both quantisation and compression artefacts are shown in figure 7.2 which is a small crop from a master dark frame image taken with different compression and multi-frame averaging settings as described in the figure legend.

Using the YUYV camera stream will avoid (M)JPEG compression artefacts but not the quantisation artefacts. Both these artefacts (but especially grey level quantisation artefacts) can be greatly reduced by the use of multi-frame averaging instead of taking just a single frame (figure 7.2). Multi-frame averaging also reduces random noise that contaminates all camera signals – and especially so in dark conditions.

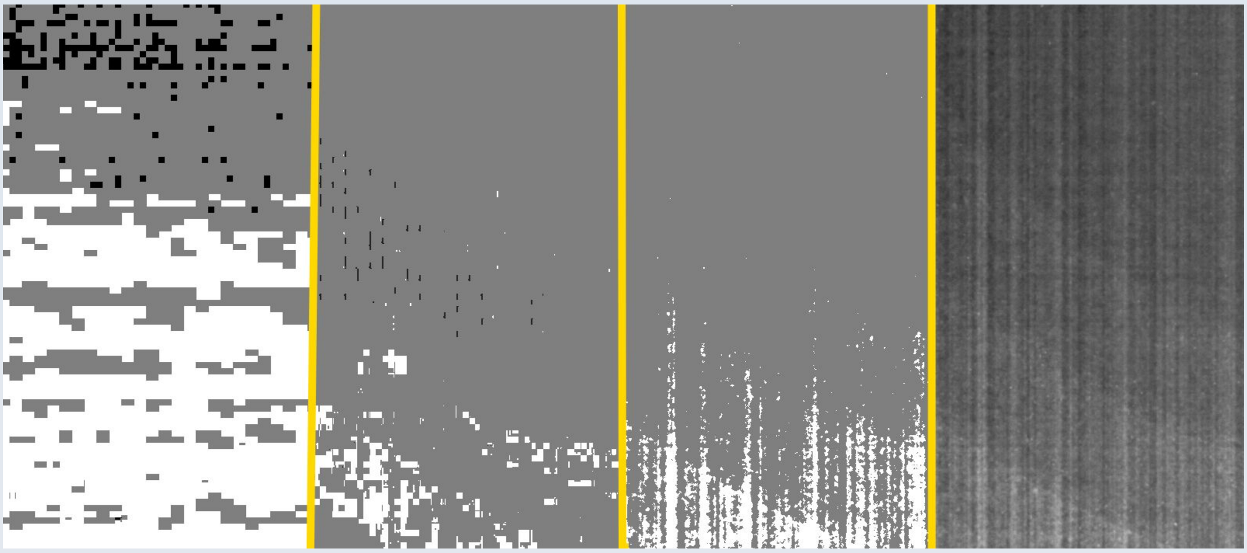


Figure 7.2 Reducing quantisation and digitisation artefacts by multi-frame averaging – all pictures are a zoomed in portion of a full image frame from a dark field image acquisition experiment with the AF51 camera, all taken from the same area and contrast is stretched for the purpose of this illustration. From *left* to *right*: Single frame MJPEG stream saved as a JPEG image (both compression blockiness and quantisation artefacts are prominent), 64-frame average MJPEG stream saved as a JPEG image (compression artefact blocks are reduced in size but the range of visible grey levels is still very limited), 64-frame average from the uncompressed YUYV image stream saved as a BMP image (compression blockiness is gone – we can resolve single pixels – but the limitation on the number of grey scales, the quantisation artefact, remains because this image was saved in an 8 bit format, 64-frame average from the uncompressed YUYV image stream saved as raw doubles showing a much smoother gradation of grey levels at single pixel resolution – the compression artefacts have been eliminated and the quantisation artefacts have been greatly reduced (the amount of reduction of quantisation artefact, just like the amount of reduction in random noise, will increase as we increase the number of frames we average).

Dark current signal, bias and amplifier patterned noise

The presence of a fixed pattern of pixel readout information in total darkness is common to CCD and CMOS cameras and seldom are pixel values ever 0 even in complete darkness. For this reason, for quantitative imaging, it is recommended to subtract a master dark frame from all captured image frames prior to any other processing. The PARD Capture software allows you to do this this for each image frame as it is captured thereby making dark frame correction very easy and convenient.

Because on chip amplifier characteristics will change with overall gain and because dark current characteristics will change with temperature of the chip and exposure time, a master dark field image (‘master dark’) should be acquired using the same constant exposure settings as will be used when taking the actual pictures you are going to correct with that master dark.

As with flat field correction, multi-frame averaging and an uncompressed image stream should be used when making a master dark and you should save the result in raw doubles format (or FITS format if you want to use it in some other software). The procedure is essentially to cap off the lens of the camera to avoid any light getting in and then collect the multi-frame average image at a particular constant exposure setting. Dark field correction comes before all other processes so no flat field correction or masking is done while capturing the master dark frames and you should not use the ‘Scale mean of each frame to first?’ option because there should be no light fluctuations between frames (which is what this setting standardises for).

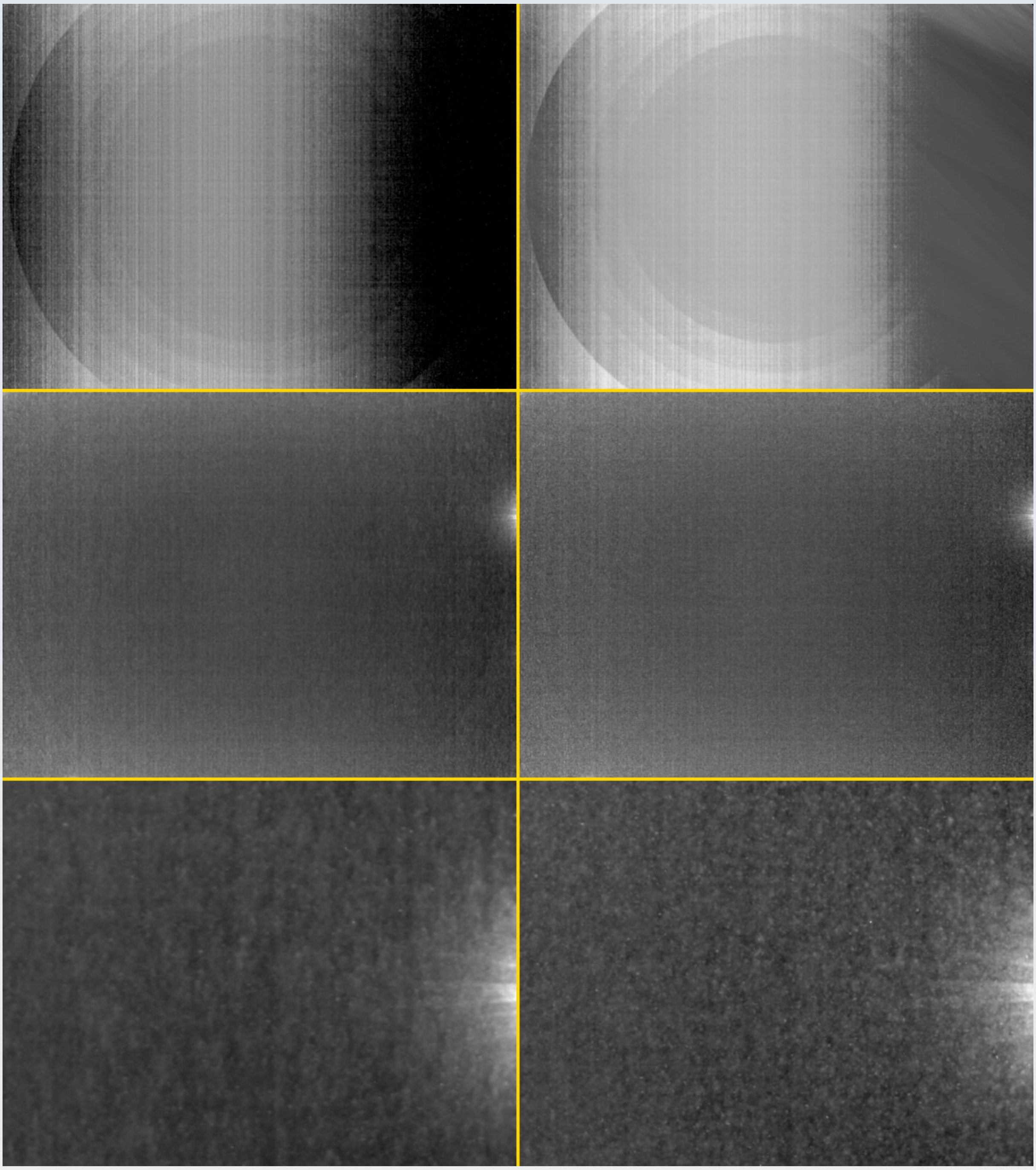


Figure 7.3 Some master dark images (each is a 64 frame average) taken with the AF51. On the *left* of each row the acquisition used the Y-channel of the YUYV camera stream. On the *right* of each row the acquisition used the Intensity channel from a YUYV-to_RGB transform still using the YUYV output from the camera. *Top*: Exposure setting of 1269. *Middle*: Exposure setting ‘night mode’ maximum frame integrations. *Bottom*: Close up of the small flare seen in the middle panel. Such small imperfections in the pixel read out amplifiers are not unusual and this is part of the fixed pattern dark field ‘noise’ which dark field subtraction is designed to correct for. See text for further detail of what is shown here. All these images are shown contrast-stretched and shown on a square root look-up-table for ease of seeing the features.

An set of example master dark frames taken with different fixed exposure settings and acquired using the procedure just described with the AF51 camera is shown in figure 7.3.

Note the concentric and linear patterns as well as the gradation of bias signal across the field. Note also how the pattern changes for different exposure settings.

For the relatively fast exposure of 1269 (top row images in figure 7.3) there is an appearance of concentric rings that may be a reflection of the fixed pattern LENC gain correction factor discussed in chapter 3. Note how this is greatly reduced when the longer ‘night mode’ multi-frame integration (on-chip) exposure is used (middle row in figure 7.3).

Note also that for the faster exposure time (top row) there are some completely black pixels on the Y-only channel (left). These completely black pixels show no variation at all in them – they all have the exact same value of ‘16’ without any variation despite being a 64-frame average. This indicates that the amount of charge on those pixels was so low that it was below the ADC amplifier threshold so only the bias signal was output (for the AF51, the bias signal – the lowest pixel value possible in the read out image – is 16 for the Y channel of the YUYV camera stream). This occurred because at this relatively fast exposure rate those pixels did not accumulate sufficient electrons by the type the rolling shutter read out the pixel values in those pixels. Under these circumstances, no amount of multi-frame averaging in software will be able to bring out any signal in those pixels at this light level (remember that this is a dark field exposure, when the lens cap is removed we can expect some light to get to those pixels and bias them sufficiently to overcome this deficiency and read out some meaningful signal).

However, when a longer exposure time is allowed such as in the middle row pictures of figure 7.3, sufficient charge has built up in all pixels across the field, even in complete darkness due to thermal dark current signal accumulation, that they all read out signal above the bias level. It is this property that makes the AF51 useful for imaging is low light level situations.

The middle and lower panels of figure 7.3 shows a hyper-sensitive area which may be due to small fluctuations in ADC amplifier performance and is one of the sources of dark field variation that using a master dark for field correction is designed to compensate for in quantitative work. The lower panel shows this area in greater detail and you can see, comparing left (Y-only channel) and right (full YUYV-to_RGB conversion followed by extraction of the intensity channel from an RGB-to-HSI transform) that we get slightly sharper resolution with less vertical smearing using the full YUYV data to derive our intensity information as opposed to using just the Y-values alone – but at the expense of some increased noise fluctuations. Using more frames in the multi-frame averaging process can reduce random noise however (I only used 64 frames for these pictures).

Colour streak artefact in low light conditions

In very low light levels with sufficiently fast exposure times such that pixels cannot read above the ADC amplifier threshold, the AF51 camera shows a particular streaking artefact where triangles appear to streak from bright pixels a bit like a tail behind a comet. This is shown in the dark field images taken with the full YUYV colour signal from the camera in figure 7.3 top, right.

This is a feature of the on-chip colour processing of the camera and does not originate in MJPEG compression because the artefacts occur with the raw uncompressed YUYV image stream as well as if the MJPEG were to be used.

This artefact only manifests in regions where no meaningful signal can be obtained because exposure is such that the bias level of the ADC is not exceeded. As shown in the middle row of figure 7.3, when a sufficiently long exposure setting is used so as to allow all pixels to be above the bias level upon readout, the artefact no longer exists.

Making a master flat and master dark for astronomy

Mean averaging (such as is done within the PARD Capture software) is sufficient for making a master dark and master flat for use with microscopy work. However, for other types of application (such as in astrophotography) other image combination approaches may be preferable. Such detailed guidance is outside the scope of this user manual but there are various texts and free web tutorials on the subject. Keep an eye on the PUMA Microscope YouTube channel for video tutorials as well. Some introductory links that may be useful in this regard are as follows:

http://spiff.rit.edu/classes/phys445/lectures/darkflat/darkflat.html

https://core.ac.uk/download/pdf/37772805.pdf

https://www.cfht.hawaii.edu/~baril/Pyxis/Help/flatdarkfield.html

https://www.stsci.edu/instruments/wfpc2/Wfpc2_dhb/wfpc2_ch44.html

http://astro.corlan.net/gcx/html/node7.html

https://github.com/deepskystacker/DSS/releases/tag/4.2.2-Release

Note that the ‘Save as FITS?’ option in PARD Capture exists so you can capture individual frames using PARD Capture and transfer them to other processing software if you wish to use the more elaborate image combination processes discussed in some of the above references.

Unnatural edge sharpening

The AF51 has a ‘Sharpness’ control which is default set to 50 (range 0 to 100). Use of this default setting can often make images seem crisper for casual illustration purposes however a high level of sharpening also brings with it unnatural ringing or overcompensation at sharp edges (figure 7.5) that can cause excessive noise or otherwise be detrimental to certain image processes (such as deconvolution). For scientific imaging therefore it is often best to reduce this sharpness setting to zero (or near zero) so that edges appear more ‘natural’ even if a little soft.

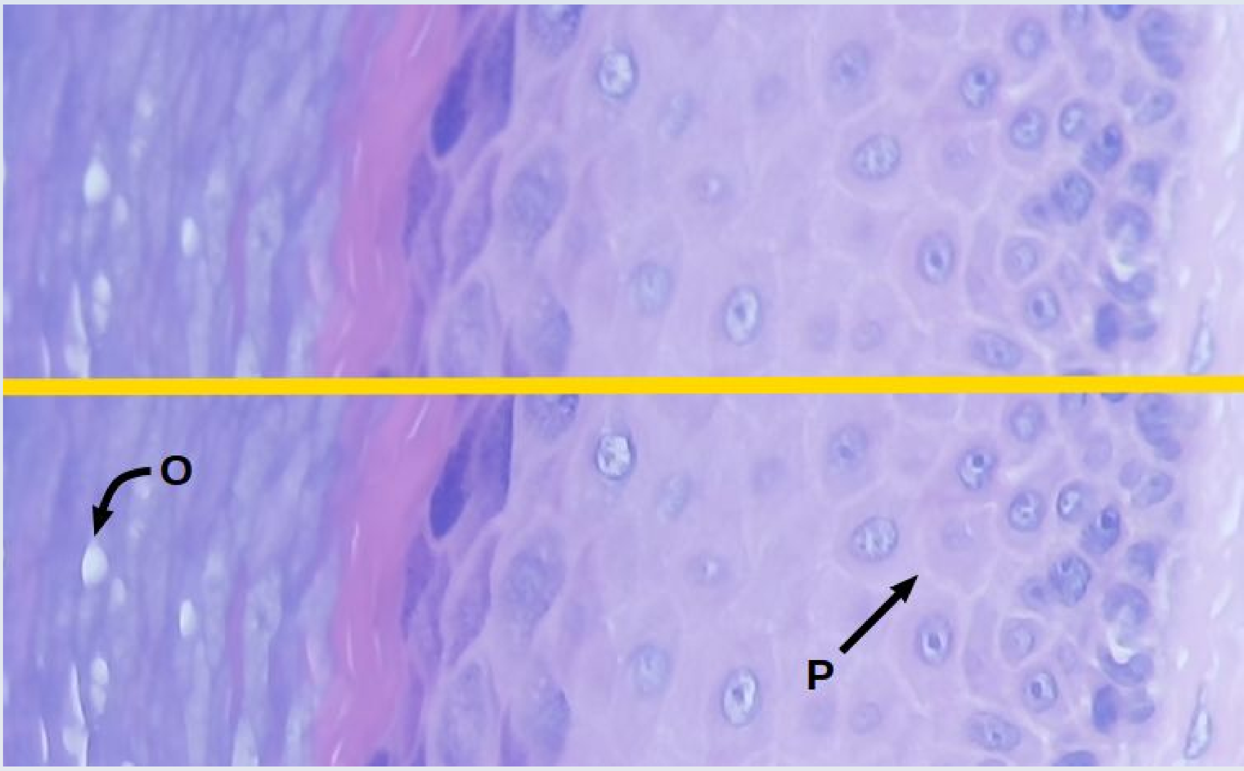


Figure 7.5 The effects of sharpening setting on the AF51. This is a small crop from a full field image of human skin biopsy specimen taken with a PUMA microscope and an OptArc x40 objective. *Top*: Sharpening factor set to 1. *Bottom*: sharpening factor set to 50. While an increased sharpening factor makes some details clearer such as the prickles between the epidermal cells (P), other structures such as nuclear chromatin and the nuclear 'holes' in the stratum corneum (O) show unnatural exaggeration of contrast at the boundaries which is not good for many types of image analysis.

Sample images

All the images in this PDF document are either downsampled or compressed or both so cannot reflect the true quality of images from the camera. A selection of full and uncompressed images taken with the AF51 camera is available on the PARDUS project GitHub page. For example, see:

https://github.com/TadPath/PARDUS/blob/main/PARD_Capture/SampleImages_AF51.md

Anyone considering purchasing or using the AF51 camera should take the opportunity to view those sample images to assess whether the quality of image output will be suitable for their purposes in addition to the information presented in this manual.

8. Disposal and Recycling

The optics are made of glass and the heat sink, lens enclosure are made of aluminium and steel.

The metal fixings are made of steel (mostly stainless steel but some elements may be galvanised steel).

The 3D printed plastic parts are made with poly-lactic acid (PLA) plastic. The following advice is current as of 2021. Consult your local authorities for the latest situation.

Recycling of PLA Plastic in the UK

PLA is compostable but most councils in the UK do not accept it in green or food waste.

PLA can be recycled but only by a very few specialist facilities. Most councils in the UK will not accept PLA plastic in their recycle bin waste.

PLA should therefore be disposed of in general household waste or sent to a specific facility that will accept it for recycling. Those with the ability to do so may also grind used PLA and reform it into usable 3D printer filament.

Recycling of PLA Plastic outside the UK

Please consult with your local authorities for recycling advice.

Electronics

The interface board and CMOS sensor contain surface mount electronic components and connectors including semiconductor parts.

Electronic components should NOT be disposed of in the general household waste or general household recycle bins. They must be disposed of in special electrical goods recycle facilities. These may be provided by your local authority is civic amenities sites. Please consult with your local authority guidelines on the disposal of electrical goods.

Index

[End of Document]