

Major Classes in FCamera

ExtendedSettings (ExtendedSettings.h)

This is a QWidget that holds all the persistent settings for FCamera. It's also a good place to put camera settings that aren't changed frequently enough to warrant going on the viewfinder. The settings are saved in the UserDefaults singleton instance (UserDefaults.h), which is backed by the file ~/.fcamera/settings. If you want to add more settings, put your code in this class. Rather than talk to this widget directly, the rest of FCamera retrieves these settings from the UserDefaults instance.

CameraParameters (CameraParameters.h)

The CameraParameters instance is owned by the CameraThread, and represents the current state of the camera. It has one sub-struct per major camera parameter (such as white-balance), which each typically contain a mode, and then whatever values are relevant to that parameter (e.g. manual white-balance needs a color temperature).

The CameraParameters instance is written to by the Viewfinder when the settings tree is used, and read by the CameraThread, which effects the changes. It is also written to by the camera thread if any parameters are in "auto" mode, as it is the camera thread which runs auto-focus, auto-exposure, and auto-white-balance algorithms.

Viewfinder (Viewfinder.h)

The Viewfinder is the main screen of FCamera, visible when you start it up. It manages a tree of settings widgets on the right (SettingsTree.h), which make sliders or other AdjustmentWidgets (AdjustmentWidget.h) appear when appropriate. The live viewfinder itself is a hardware YUV overlay (OverlayWidget.h) - all Qt does here is draw a rectangle of constant color, which is treated by the hardware as a color key to draw YUV video on. Finally, the Viewfinder shows any active visualizations, like histograms, on top of the overlay (VisualizationWidget.h).

ThumbnailView (ThumbnailView.h)

This is the screen you review your photos on. It has an HScrollArea on the left that contains QLabels showing recent images. You can slide it left and right to browse your image gallery. All images are stored as RAW (in the .dng format), and are demosaicked as needed for display.

On the right are various actions you can take. If you zoom, a ZoomableThumbnail (ThumbnailView.h) demosaics the current image at full resolution, and appears over the HScrollArea to let you zoom and pan the image. The photos in the gallery are each represented by an ImageItem (ImageItem.h). All the actual work of loading and saving ImageItems is done by the IOThread (ImageItem.h).

CameraThread (CameraThread.h)

The CameraThread talks to FCam and controls the camera. Its actions are guided by the values it reads from the CameraParameters instance. Most of the time it's running a viewfinder, and is set up to place the image data directly into the OverlayWidget's framebuffer. When it takes a photo, it creates a new ImageItem to represent that photo, and passes it to the ThumbnailView to deal with.

VScrollArea (ScrollArea.h)

This container widget displays one of its children at a time, and lets you slide between its children by swiping your finger. It acts much like the desktop background on the N900. This one is vertical, and contains the three main views of FCamera - the ExtendedSettings at the top, then the Viewfinder in the middle, and finally the ThumbnailView down below. The other place in FCamera a ScrollArea is used is on the ThumbnailView.

ParameterButton (SettingsTree.h)

Only one may be active at a time. When active, it shows its child ModeButtons. It also displays the current mode that parameter is in, which it gets from the CameraParameters instance.

ModeButton (SettingsTree.h)

Only visible when its parent ParameterButton is active. May optionally show an arbitrary child widget when active (usually an AdjustmentSlider or AdjustmentTouchArea (AdjustmentWidget.h)). When a ModeButton is selected, or one of its child widgets is changed, they notify the Viewfinder, which extracts the new information and updates the CameraParameters instance.

