

PrimeLocate™

An excess of image data may seem like an abundance of riches, except when the data does not contain useful information. Then it becomes data that must be stored, processed and ultimately discarded. The Prime™ family of cameras from Teledyne Photometrics solves this problem at its source with PrimeLocate, one of its advanced, real-time processing features. When enabled, PrimeLocate automatically finds and transfers regions of interest to the host computer, either over PCIe or USB3. In this way massive amounts of data containing no useful information is eliminated from consideration. *PrimeLocate can identify up to 512 of the brightest 3x3 patches within each image and transmit only that portion of the image, reducing the data and processing requirements from 6.7Gbps down to 100Mbps.*

The PrimeLocate algorithm is most appropriate for localization microscopy methods such as STORM and PALM. The hallmark feature of localization microscopy is that sparse images of individual point emitters blink at random times during an image sequence. By finding the centroid of each emitter's diffraction limited spot in a given frame and combining the localization results from each frame, a super-resolution image of the original fluorescence can be reconstructed.

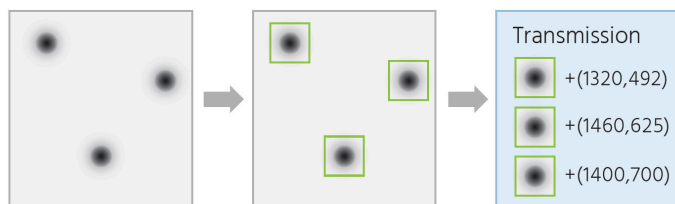


Figure 1. The PrimeLocate process in the camera. From left, an image containing sparse events is analyzed. The regions containing these events are identified, and only those regions are transmitted. Coordinates of the regions are also transmitted as metadata attached to each region.

The PrimeLocate algorithm scans each full image (or single ROI, if enabled) for the brightest local maxima within a 3x3 window. Each identified bright local maxima forms the center of a transmitted patch. The transmitted patch size is under user control and can vary from a radius of "1" to transmit a patch of 3x3 pixels, to a radius of "15" for transmitting a patch of 31x31 pixels. The number of transmitted patches can vary from one to a maximum of 512 patches per frame.

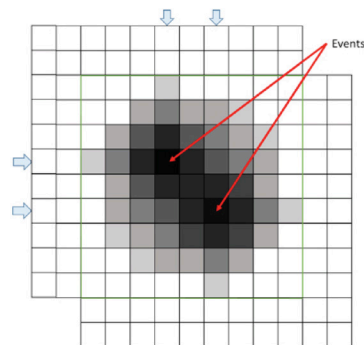


Figure 2. Example of PrimeLocate for two partially overlapping fluorescence events. The patch maxima are separated by more than two pixels, so two patches are transmitted with partially redundant pixel data. This ensures all data needed for correct localization is included. The blue arrows mark the center rows and columns for the patches.

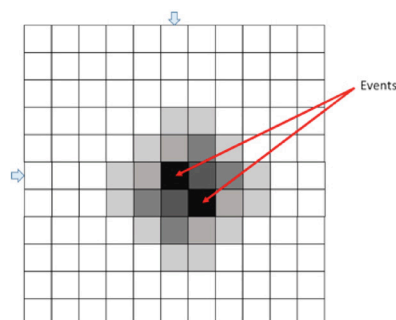


Figure 3. Example of 11x11 patch data for two small, overlapping fluorescence events. The event maxima are close enough together that a single 11x11 patch captures both. The blue arrows mark the center row and column for the patch.

Using a small 3x3 search window ensures that when two fluorescence events happen in close proximity, two or more patches are transmitted to capture both events (Figure 2). The 3x3 exclusion region discounts two or more events occurring within a radius of two pixels as being independent, as events this close together can be captured within a single, transmitted patch (Figure 3).

PrimeLocate always transmits the predetermined number of patches per frame, regardless of the number of fluorescence events present. For example, when using the maximum 512 regions, after patches containing true fluorescence events are transmitted, remaining patches will simply contain the brightest noise peaks from the

image. In this way, a consistent amount of data is transmitted per frame. These patches will be low-intensity compared to the brightest events and can be ignored. Should more than 512 events occur within a single frame, those events will be lost.

PrimeLocate does not attempt to localize events to precision better than a pixel, as this step is expected to be performed on the host. Leaving this step to the host allows choice of localization algorithm and configuration. The data transmitted to the host includes the patch data itself as well as the patch's (X,Y) location, so that the sparse image can be reconstructed at the host.

PrimeLocate and µManager

µManager is an excellent open-source microscope control application that provides image acquisition and device control (www.openimaging.com). µManager supports PrimeLocate along with several new innovative features of the Prime family of cameras. Settings for PrimeLocate are found in the "Device Property Browser...", or as shown below, the "Configuration settings" dialog when using the Hardware Configuration provided with the Prime and Prime 95B cameras.

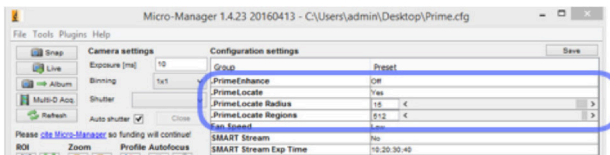


Figure 4. PrimeLocate settings in µManager

There are three highlighted controls:

Name	Description	Selection
PrimeLocate	Controls if PrimeLocate is enabled	Off/On
PrimeLocate Radius	Controls the size of the region transferred around each local maxima	1 to 15
PrimeLocate Regions	Controls the number of maxima transferred	1 to 512

Table 1. Highlighted Controls.

In the following example, a 512x512 portion of the sensor was used to acquire 1000 frames at 300 FPS. In each frame, 16 regions were selected for transfer using PrimeLocate with a size of 31x31 pixels (radius=15).

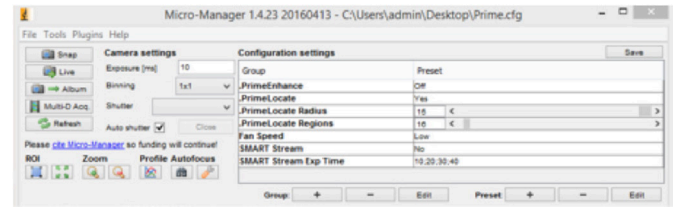


Figure 5a. PrimeLocate settings in µManager.

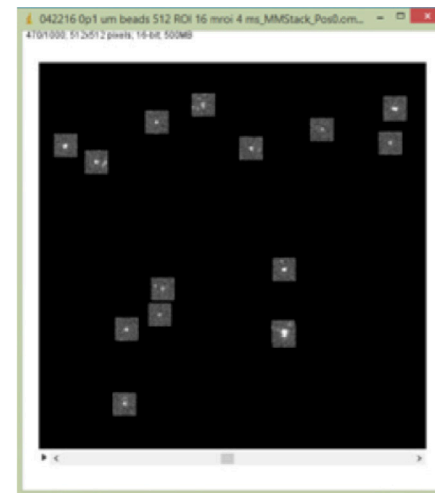


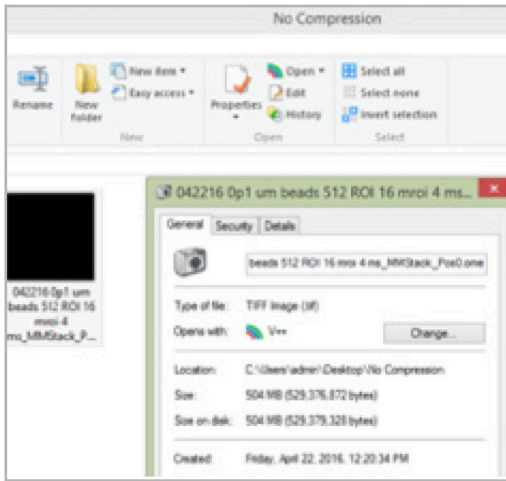
Figure 5b. Example of a 512x512 region, with PrimeLocate configured to find the 16 brightest regions, and return them as 31x31 pixels patches.

PrimeLocate and Data Reduction

Windows provides a convenient method for taking advantage of compression by saving images to a "compressed" folder on the hard disk. Windows will dynamically compress the image data as the images are written.

An uncompressed TIFF format image stack consisting of 1000 frames of 512x512, 16-bit image data without PrimeLocate enabled is 504MB uncompressed, and 346MB when saved to a compressed folder.

However, when using PrimeLocate configured for 16 regions of 31x31 pixels, the same file consumes only 54MB in the compressed folder as the blank-zero valued area will now efficiently compress using Windows compression algorithm.



When using the maximum 512 regions, more data from the original image is transferred, and this same file is reduced to 257MB. These results are typical, the exact amount of compression depends on the image data and the number and size of stored regions.

Conclusion

This brief introduction is designed to provide background on how PrimeLocate works. The next White Paper for PrimeLocate will focus on how it can be adapted for live particle tracking.

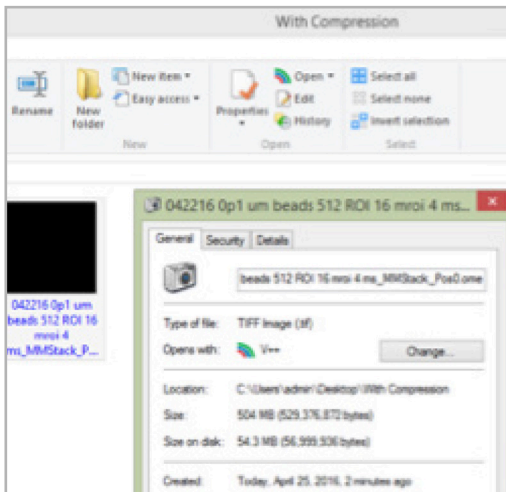


Figure 6. PrimeLocate enables effective compression using Windows compressed folder mechanism, resulting in a 10x reduction in storage requirements in the 1000 frame example shown.

