Particle image velocimetry from multispectral data

John Charonko\textsuperscript{1}, Elizabeth Voigt\textsuperscript{1} and Pavlos Vlachos\textsuperscript{1}

\textsuperscript{1} Department of Mechanical Engineering, Virginia Tech, Blacksburg, Virginia, USA
john.charonko@vt.edu

ABSTRACT
Since the adoption of digital video cameras and cross-correlation methods for particle image velocimetry (PIV), the use of color images has largely been abandoned. Recently, however, with the re-emergence of color-based stereo and volumetric techniques, color imaging for PIV has again become relevant. In this work we explore the potential advantages of color PIV processing by developing and proposing several methods for handling multi color images. The first method uses cross-correlation of every color channel independently to build a color vector cross-correlation plane which can be searched for one or more peaks corresponding to either the average displacement of several flow components using a color ensemble operation, or the individual motion of colored particles, each type with a different behavior. In the second case, linear unmixing is used on the correlation plane to separate out each known particle type as captured by the different color channels. The second method introduces the use of quaternions to encode the color data, and the cross-correlation is carried out simultaneously on all colors. The resulting correlation plane can either be searched for a single peak corresponding to the mean flow, or multiple peaks can be used with velocity phase separation to determine which velocity corresponds to which particle type. Each of these methods was tested using synthetic images simulating the color recording of noisy particle fields both with and without the use of a Bayer filter and demosaicing operation. It was determined that for single phase flow, both color methods decreased random errors by approximately a factor of 2 due to the noise signal being uncorrelated between color channels, while maintaining similar bias errors as compared to traditional monochrome PIV processing. In multi-component flows, the color vector correlation technique was able to successfully resolve displacements of two separate flow components with errors similar to traditional grayscale PIV processing of a single phase. It should be noted that traditional PIV processing is bound to fail entirely under such processing conditions. In contrast, the quaternion methods, frequently failed to properly identify the correct velocity and phase and showed significant cross-talk in the measurements between particle types. Finally, the color vector method was applied to experimental color images of a microchannel designed for contactless dielectrophoresis particle separation, and good results were obtained for both instantaneous and ensemble PIV processing. However, in both the synthetic color images that were generated using a Bayer filter and the experimental data, a significant peak locking effect with a period of two pixels was observed. In order to mitigate this detrimental effect it is suggested that improved image interpolation algorithms tuned for use in PIV are applied on the color images before processing, or that cameras that do not require a demosaicing algorithm are used for PIV.

INTRODUCTION
In current practice, particle image velocimetry (PIV) typically uses monochrome digital video cameras to capture images of a displacing particle image pattern, often at very high frame rates. With these cameras, the full multispectral information from a scene is compressed into a single intensity channel, and only through the use of color filters can any determination of the original wavelength of the incident light be made. This however, is not typically seen as a problem, as current PIV algorithms are only designed to work with a single channel of intensity data at a time. This is often sufficient since the particle images are typically illuminated with a single wavelength laser beam, and the particles either reflect that single color of light, or excite a single color of fluorescent particle to emit another narrow band of wavelengths. If multiple color particles are used, separate cameras with tuned wavelength filters are set up to image each one. For color images, the images must first be converted into a single scalar value, or a single color channel must be isolated for processing.

Early PIV systems, before the advent of digital video recording and cross-correlation [1], often used photographic film to capture particle images. In these systems, which predated the widespread availability of hardware capable of cross-correlation PIV, recordings were typically made with multiple exposures on a single frame, and the particle images were autocorrelated to yield the displacement information [2-4]. However, this left the user with a directional ambiguity. One solution to this problem was to take advantage of the inherent color sampling properties of the photographic film and record two light pulses of different colors [5, 6]. The color recordings were often then digitized using a second digital camera and matching optical filters to scan and process the developed photographic negative. This would result in a pair of images, each with only a single particle exposure, which could then be paired using particle tracking or some variant of image correlation. Using this method, the color information played an inherent part
in a successful PIV measurement, but in the final analysis each image was still handled by the matching algorithms as a monochromatic scalar intensity field. However, as digital video cameras and cross-correlation PIV became more mature, such approaches largely fell by the wayside due to the large benefits in efficiency provided by all-digital systems.

There is also a long history of using spatial variation of color to assist in the determination of a third out-of-plane velocity component. Early approaches used a pair of overlapping and differently colored laser beams of Gaussian intensity profile [7, 8]. The out-of-plane component could then be recovered through manipulations of the relative intensities between the two planes since the particles in the overlapped region would have varying contributions of each color based on their depths. More recently, these techniques have been extended to multiple distinct light sheets, in which in-plane velocities are tracked by separating the recorded color images into multiple channels and the out-of-plane components are recovered from standard two-camera stereo methods [9], or to systems in which the color sheets are overlapped [10] or continuous across the volume [11] so that velocity information can be recovered by identifying the color (and thus the depth) of each particle, with particle tracking velocimetry algorithms used to determine the full volumetric flow field.

Nevertheless, most of the existing work has treated the color information simply as a way to sort the various particle images into different categories, and has ignored its potential use as a way to increase the information content of the PIV recordings, and thus the signal to noise ratio. To do this, in this work we preserve the information from all color components throughout the cross-correlation analysis rather than simplifying to a composite intensity channel or examining only a single correlation between two pre-defined channels. Additionally, this work proposes two methods that have not been previously applied to PIV processing for using color images to simultaneously sort and measure the velocities of two or more types of particles in multiphase flows. Although color images have been previously used to classify simultaneously measurements of particle velocities in multiphase flows, these methods typically rely on a series of multiple filters, light sources, or cameras, and complex data reduction schemes to delineate the behavior of each particle type [12]. In contrast, the approach presented here is more straightforward and can be performed with the simple introduction of a color camera into a standard digital PIV setup.

These techniques are based on two distinct approaches to handling the color information. The first is the correlation of all available color channels as distinct planes, and the assembly of those correlations into a color vector cross-correlation plane. This plane can then be searched for the peak value in an average sense if only a single velocity component is desired, or for peaks matching the individual colors of the particles which were imaged. It is not necessary, as has been done previously, to carefully choose particles whose colors fall very nearly into a single color channel on the camera sensor (typically a particular wavelength of red, green, or blue).

The method for identifying a signal of a particular color in a mix of multispectral sources is implemented using a technique known as linear unmixing [13]. It originally grew out of work on hyperspectral data analysis in the satellite imaging community [14] and has recently been adopted by the biological imaging community for identifying cells stained with or expressing various fluorescent pigments [15, 16]. It is designed to separate features most closely matching a set of reference colors from data containing a mixture of recorded wavelengths. However, instead of applying unmixing on the recorded images, here unmixing will be applied to the color vector correlation planes.

The second approach is the encoding of the three component color information into a type of number known as a quaternion, and then processing the images simultaneously across the entire recorded colorspace in a single operation. Quaternions are a type of hypercomplex number first proposed by Hamilton in 1843 [17], which encode four dimensions of real scalar data into a real part (w) and three imaginary parts (a, b, and c), e.g. \( q = w + ai + bj + ck \) where \( i, j, \) and \( k \) are orthogonal, imaginary components such that \( ii = jj = kk = -1 \) and \( ij = k, jk = i, ki = j \). In the case of color image data, it is common to decompose the perceived color of a given light source at each location in an image into three components, e.g. the RGB (red, green, blue) colorspace. Thus, for RGB images, it is natural to encode this color data as the \( i, j, k \) components of a pure quaternion 2D matrix.

\[
I(x,y) = R(x,y)i + G(x,y)j + B(x,y)k
\]

Once color vector data have been encoded into a 2D or 3D matrix of quaternion values, the standard Fourier Transforms (FT) in both discrete and continuous forms can be extended to apply to quaternion fields. Sangwine, Ell, and collaborators [18, 19] have shown that the single forward and inverse Fourier transforms for real or complex numbers give rise to families of related quaternions transforms. One such family is shown in Eq. (2).
The scalar color are summed, and individual color correlation in combination with linear including information from three scalar color images. Their relationships are shown schematically in Figure 1; each of these methods is described in more detail below. In each case, the methods were adapted to allow the use of a phase-only transform and the spectral energy filter of the RPC (Robust Phase Correlation) method [24].
Figure 1: Schematic of the three proposed color cross correlation techniques. Color ensemble correlation averages the scalar correlation of each color channel into a single field. Linear unmixing uses knowledge about the color of each source particle to separate the full RGB correlation plane. Quaternion cross correlation operates on RGB data stored as hypercomplex numbers, and is then used to separate the source images based on the velocity of each particle, rather than the color (mean RGB values of the separated images are used to sort the measured displacements).

Quaternion Cross Correlation

For this approach, diagrammed in the lower left corner of Figure 1, each RGB color image was encoded as a 2D quaternion matrix using eq. (1). These quaternion images were then cross-correlated using the discrete form of eq. (3). Before leaving the frequency domain, the phase-only conversion was applied by dividing by the magnitude of each spectral component of $C$. To implement the RPC, the spectral energy filter was encoded into the scalar part of a quaternion array, and multiplied element-by-element into the phase-only cross-power spectrum. The correlation was then completed using an inverse Fourier transform. The result of this operation is depicted as $C_{Q}$ in Figure 1, with the real part giving the magnitude of the correlation surface, and the vectors depicting the direction of the three hypercomplex components of the quaternion field.

As previously stated, the cross-correlation that results includes both real and hypercomplex components. This creates an ambiguity about how to define the maximum value for the peak search, and what values to use in a subpixel fit (here implemented with a standard three-point Gaussian fit in each direction). The two obvious approaches are to either use the magnitude of the resulting quaternion number, or the absolute value of the real part. Based on preliminary testing (results not shown) it was determined that the second approach yields slightly better results, and was therefore used for the remainder of this work. It is conjectured that this is because the magnitude could be contaminated by correlations resulting from matches of different color particles (which would be away from the real axis). The scalar component should correspond primarily to color-matched translated images, and should be less affected by cross-color particle matching; however, this has not been examined in detail.

In the case of multiple displacements of different colors, since a translation of matching color elements creates a scalar peak regardless of the color of the translating image components, an additional step is required to sort out the source color of the particles that give rise to each peak in the correlation plane. The procedure used for this is described in Velocity Phase Separation section below.

All quaternion math was carried out using version 1.9 of the Quaternion Toolbox for MATLAB (qtfm), available on SourceForge [25].
**Color Vector Cross Correlation**

As an alternative to using quaternion cross correlation, it is also possible to use traditional scalar methods on each separate color component, yielding three separate correlation fields, \( C_R, C_G, \) and \( C_B, \) respectively for the red, green, and blue channels. This can be accomplished using any cross-correlation scheme the user desires, though as previously stated an RPC method was employed in this work. These three correlation planes can then be combined into a color image \( C_{RGB} \) where the color of each peak corresponds to the average color of the correlating structures that gave rise to it. For the unlikely case where each individual flow tracer is visualized by only a single color channel, the displacements can be determined by simply finding the peak in each channel and applying a subpixel fit. For example, for microPIV applications this would require the use of fluorescent probes with emission spectra matched to the camera color filter with no overlap between filters. However, this is not typically the case and each flow tracer is often visualized by color components in each of the three channels, giving rise to correlation peaks in all three correlation planes (as shown in Figure 1). This is in contrast to quaternion cross correlation, in which the peak is in the scalar direction if the color of the object is the same in both frames, regardless of the original flow tracer color. This provides a clear advantage for color vector cross correlation if the approximate color of the flow tracers is known ahead of time.

To separate out the contribution to the correlation plane of each particle color, linear unmixing (described in detail in the corresponding section below) is applied to the \( C_{RGB} \) correlation plane, resulting in scalar correlation planes corresponding to each original particle type (planes \( C_{P1}, C_{P2}, \) and \( C_{P3} \) in the lower right of Figure 1) with minimal cross talk from particles of different colors. The peaks are then identified and a standard Gaussian three-point fit is used to locate their subpixel locations.

**Color Ensemble Cross Correlation**

For the case in which all tracers have uniform motion, it is not necessary to consider each color channel separately. Traditionally, if color images were used to image such a flow, these images would be converted to grayscale before cross-correlation, or only a single color channel would be chosen. Here it is proposed that the full color information be kept until after the correlation step in order to preserve the additional information available from the original images. At this point, the three-color correlation fields are averaged to yield a scalar field in which the peak could be located using standard PIV techniques.

\[
C_{(RGB)} = \frac{1}{3}(C_R + C_G + C_B)
\]  

(4)

This technique is most appropriate where the group velocities of all tracer types are approximately equal. For the case of different particle colors with different motions, the source of the peaks are not distinguishable based only on information in the correlation plane. As proposed previously, it should be possible to use velocity separation to decompose the original images into components corresponding only to each of the identified motions.

**Linear Unmixing of Cross Correlation**

Linear unmixing was performed by solving the following inverse problem[13].

\[
\{ S(\lambda_i) \} = [ R(\lambda_i) ] [ A_i ]
\]

(5)

\( S \) represents the signal as a function of the sampled wavelengths, \( \lambda_i, \) and \( R \) is a matrix of the linear response of the sensor at each wavelength to the excitation of each of \( A_i \) input types. In this case, our signal wavelengths are limited to the red, green, and blue color channels of the digital color image, though in a general case many more wavelengths can be sampled using different filters. The input types \( A_i \) correspond to different particle types used in the images. The number of different source types must be less than or equal to the number of signal channels for the problem to be solvable. Given these restrictions, for the unmixing problem studied here, eq. (5) can be written more specifically as follows.

\[
\begin{bmatrix}
S_R \\
S_G \\
S_B
\end{bmatrix} =
\begin{bmatrix}
R_{p1}(R) & R_{p2}(R) & R_{p3}(R) \\
R_{p1}(G) & R_{p2}(G) & R_{p3}(G) \\
R_{p1}(B) & R_{p2}(B) & R_{p3}(B)
\end{bmatrix}
\begin{bmatrix}
A_{p1} \\
A_{p2} \\
A_{p3}
\end{bmatrix}
\]

(6)

In the presence of noise and other errors in the measured signal, an exact solution was generally not available, and instead the relative intensities of each component \( A_i \) were solved for by a least squares inverse algorithm. When only two source components were used to form the final images (i.e., two particle types) the response matrix coefficients...
corresponding to the third component were set to dummy values, thought tests revealed that they could be safely removed without affecting the final results (changing the inversion problem to an overdetermined system).

For tests dealing with synthetic images, the response matrix was determined based on the known input colors of each particle type. For experimental images, approximate values of \( R \) were determined by manual inspection of the recorded color images, but might also be recovered more accurately using knowledge of the particle emission and camera filter spectra.

**Velocity Phase Separation of Images**

As mentioned previously, due to the formulation of the quaternion cross correlation which only yields scalar peaks regardless of the color of the matched patterns, an ambiguity exists regarding which of the particle types created each of the identified peaks. To solve this problem, we decomposed particle image patterns into the sum of two or more sub-images in the Fourier domain, each with a unique associated velocity, using Eq. (7). This approach is based off the work of Alexiadis and Sergiadis, who originally described a similar procedure for use in motion tracking for machine vision [23].

\[
\begin{bmatrix}
F^{+tL}_{\{\bar{\Pi},t\}} \\
F^{+tL}_{\{\bar{\Pi},t+\Delta t\}}
\end{bmatrix} = \begin{bmatrix}
1 & 1 \\
e^{-\mu t j} & e^{-\mu (t+\Delta t)}
\end{bmatrix} \begin{bmatrix}
C^{+tL}_{\{\bar{\Pi},t\}} \\
C^{+tL}_{\{\bar{\Pi},t+\Delta t\}}
\end{bmatrix}
\]  

(7)

The equation arises from the Fourier shift theorem, which states that a shift in image coordinates is manifested by a linearly varying complex phase in the Fourier domain with a slope proportional to the shift. Given that \( F \) and the displacement terms are known, inversion can be used to determine the \( C \) terms one frequency component at a time. Before the inversion is computed, the discriminant of the shift matrix is computed and any matrices that are too close to zero are removed without affecting the final results (changing the inversion problem to an overdetermined system).

After separating the subimages into the two individual particle images at time \( t \), the flow tracer matching each of the velocities that was used to perform the decomposition was identified by inspection. For this work, that meant calculating the mean color of the remaining particles in each image and determining which particle each most closely matched. However, other criteria could be used; in particular a sorting algorithm similar to that used by Khalitov and Longmire [26] using the particle size and intensity could prove useful in multiphase flow experiments.

If desired, eq. (7) can easily be extended to single channel greyscale images as a special case in which the Fourier transforms become their familiar complex number forms, and \( \mu = -i \). In this form, such a procedure could have also been applied to the color ensemble processing to resolve more than a single velocity component, as was previously discussed.

More than two separate translations can also be handled by adding additional images at different time instants (\( t+2\Delta t \), etc.) and displacement terms (\( v_j \), etc.), though initial testing suggests that the robustness of the method decreases quickly with increasing number of images. This is partially because the velocity tends not to remain constant between frames, and partially because it becomes more difficult to correctly identify additional peaks in the correlation plane as the number of peaks increases. This weakness is in contrast to linear unmixing, which is only limited in the number of particle types that can be resolved by the number of individual color channels that can be acquired, though in practical terms for RGB images the limit is 3.

To apply this method to PIV data, we first divided each flow field into smaller interrogation regions, which were processed using quaternion cross-correlation, and subsequently identified the two largest peaks in the resulting correlation plane. Then, each ROI was decomposed using eq. (7) into two subimages, each matching one of the identified velocity peaks. As previously mentioned, each subimage was sorted based on the average particle color and a final flow field was assembled for each tracer type.

Additionally, due to the slow performance of the inverse operator for quaternion numbers as implemented in the QTFM toolbox in MATLAB, it was determined that equivalent results could be obtained for color images by separately decomposing each color channel of the original images using the scalar form of eq. (7), and then recombining the resulting subimages into full color images. This resulted in dramatic time savings in the final implementation of the algorithm.
Synthetic Image Generation

In a physical camera, most digital cameras record color images in one of two ways. In the first, incident light is split using filters (either inherent to the sensor or separate) into at least three distinct wavelength bands, and three or more sensors image every color across the entire scene. These sensors can either be stacked in alignment with the light path (such as the Foveon X3), or split up with the light being directed using prisms or mirrors (so-called “3CCD” or “3MOS” designs). Usually the colors correspond to the red, green, and blue wavelengths typically used in storing or transmitting RGB image data. The second, more common approach is to place a filter at each pixel on a single sensor so that each pixel accepts light from only a single color band, again usually corresponding to either red (R), green (G), or blue (B) light. The most common pattern for such a filter is known as a Bayer filter, and the colors are arranged in a repeating 2x2 mosaic pattern with G and R alternating in one row and B and G in the next. Under this arrangement there are twice as many green pixels as either red or blue. Other arrangements of color filters are possible, but are less common. The full color image with RGB data at each pixel must then be reconstructed using a demosaicing algorithm, and often a secondary de-aliasing filter is applied to minimize image color artifacts caused by this approach, although this typically has the effect of reducing resolution as well. Full color images can also be reconstructed by sub-sampling the original sensor data so that each reconstructed pixel contains data from at least one complete Bayer filter element, in which case interpolation is not necessary. This reduces the possibility of color artifacts, but reduces the maximum available spatial resolution.

To analyze the performance of the new methods proposed in this paper, color synthetic images of PIV particle fields were generated to simulate the above camera imaging processes (see Figure 2). Because most color cameras use a Bayer filter, or similar color filter, to image three color data on a single CCD or CMOS sensor it was important to model the effect of this procedure on the resulting images. Additionally, for the same particle fields, images recreating the grayscale intensity field without color filtering, as well as full color images without the use of a Bayer filter were simulated. The full procedure for the image generation process is detailed below.

![Figure 2: Artificial color image generation procedure. Full spectrum color images are generated from images of single color particle fields and full spectrum and native grayscale images are recorded. For Bayer filtered images, the full spectrum image is subsampled, noise is added, and then the images are demosaiced and saved as either color or grayscale. Solid colored lines indicate paths for data of that color, while checkered lines indicate Bayer filtered data. Dashed black lines indicate the addition of noise to the signals.](image_url)
discretized to a dynamic range of 8 bits (intensity counts from 0 to 255). Particles were uniformly distributed across a Gaussian light sheet. Each of these particle image pairs was then assigned an RGB color, and the selected full-intensity color was linearly scaled by the corresponding grayscale intensity field and the resulting fields were summed to yield a three-color image (“full spectrum image, noise free” in Figure 2). Values exceeding the maximum intensity were truncated. This image formed the baseline image from which all the other versions were derived. Perspective effects on the apparent motion and location of each particle were not simulated.

To simulate intensity-only grayscale images as might be acquired by a standard high-speed digital PIV camera, this full-color image field was collapsed to a luminance-only field using the RGB2GRAY () command in MATLAB. The function calculates the luminance according to the following form.

\[ L = 0.2989R + 0.5870G + 0.1140B \]  

(8)

This form is the same as used to calculate the luminance channel for a 3-channel NTSC luminance-chrominance image. At this point, the data is assumed to be equivalent as what the sensor would be imaging, and normally distributed noise with a standard deviation of 5% of fullscale and a mean of 5% was added to the image (“raw grayscale with noise” in Figure 2).

To simulate a full spectrum image with noise (such as might be acquired by a Foveon X3 camera) the same noise field as used in the previous image was added to the green channel, and then rotated and flipped to create additional noise fields that for a given point in the image would not be locally correlated between color image channels.

To generate Bayer-filtered color images, the noise free full color images were subsampled according to a GRBG Bayer filter to create a mosaic single-sensor intensity image. At this point, the noise field used in the grayscale images was added to the synthetic sensor data. The resulting image field was then demosaiced using MATLAB’s DEMOSAIC ( ) command, which implements a gradient-corrected linear interpolation. De-aliasing was not used in order to preserve the maximum possible spatial resolution. This image was then used as-is (“Bayer color with noise” in Figure 2), or was further truncated (again using RGB2GRAY ( )) to yield an image that might represent the output if a researcher were to average a color image field for processing in a traditional PIV algorithm that only deals with grayscale images (“RGB to gray with noise” in Figure 2). Other simplifications might include using only a single color channel or averaging the intensities of each channel, but were not tested here.

It can be seen qualitatively from the final images in Figure 2 that images which went through the Bayer filter / demosaicing steps show increased color and intensity noise in the resulting images. As will be shown later in the Results section, this has an impact on the use of these images for PIV processing.

**Artificial flow fields**

To evaluate the various standard and proposed PIV algorithms, the synthetic image generation procedure described above was used to simulate recordings of several different simple flow fields. For each image pair and particle type a flow field was generated, and particles were randomly seeded throughout the image. The particle locations were then displaced in accordance with the simulated velocity field to yield an image in which the displacements were a function of the particle type. In each case it was assumed that the simulated flow tracers would be independent of the motion of any surrounding tracers.

For each flow field, an RPC-based correlation was used as the cross-correlation kernel [24, 27], and each region of interest was 64x64 pixels in size and windowed down using a Gaussian function to an effective circular resolution of 32 pixels [28]. Images were 1024x1024 pixels in size for each case. The flow field was sampled on a 32x32 pixel grid for 0% overlap to maintain independent measurements, for a total of 1024 vectors per synthetic flow field.

Only uniform flow was simulated, under the assumption that the relative effects of shear and rotation would be approximately similar to previously tested PIV algorithms.

**Uniform flow**

The first image set was designed to test the effect of using color correlation techniques on a flow field in which the imaged particles were multiple colors but all traveling in the same direction. For this test, three different particles were simulated, all with equal velocity. Velocities ranged from 0 to 8 pixels/frame in the x-direction, and were fixed at 0.3 pixels/frame in the y-direction. The intent was to see if multispectral images of particles could improve the accuracy and precision of PIV measurements as compared to single wavelength (or grayscale) images.

Three different particle colors were simulated. Particle 1 had at 100% intensity a color of (63,255,192), or mostly green/cyan in an RGB colorspace. Particle 2 was mostly red with a little magenta, (255,15,127). Particle 3 was mostly
blue (15,95,255). These colors were chosen based on preliminary experimental images of microchannel flow with three different fluorescent particle types approximating red, green, and blue emission spectra.

Due to the chosen colors, the green particles had the largest intensity at 100% brightness, followed by the red and then finally the blue. Pure colors were intentionally not chosen in order to allow for the effect of crosstalk between color channels, since in real images it is almost never possible to pick wavelengths that only appear in one of the RGB color channels of a camera. Particles were seeded into the flow at an average density of 0.005 particles/pix² for each color, or 0.015 particles/pix² total. This yielded about 15 particles per 32x32 interrogation region.

Two component flow
The second test condition examined the ability of the multi-spectral methods to distinguish the motion of two independent velocity tracers and evaluated the error of the resulting measurements. For these image sets, only the particle 1 (green) and particle 2 (red) types were used, and the seeding density was 0.01 particles/pix² for each, for a total density of 0.02 particles/pix². It is important to note that since we were trying to resolve each group independently, it is the seeding density for each particle type that controls the success of the PIV correlation, not the total. For 32x32 pixel ROIs only about 10 particles on average are visible per type, which is on the lower end of the generally accepted optimal particle density [2]. For each image pair, each of the two particle colors was assigned a displacement between 0 and 8 pixels/frame in the x-direction. Displacement in the y-direction was set to a uniform 0.3 pixels/frame for both particle fields.

Experimental Flow in a Microchannel
Contactless dielectrophoresis (cDEP) was selected as a representative application for which this method can be used. cDEP is a recently developed technique for particle and cell manipulation and sorting. In conventional DEP, an electric field applied between electrodes inserted into a microfluidic device exerts a force on dielectric particles in the fluid. The dielectrophoretic force depends on the particle radius r, fluid permittivity εᵣ, and electric field gradient ∆E², as well as the Clausius-Mossotti factor F_CM, which describes the relationship between the dielectric constants of the particle and the fluid:

\[ F_{DEP} = 2πεᵣr³Re[F_CM]V|E|^2 \] (9)

Because of the relationship between DEP force and particle properties, DEP can be used to separate particles with different size and electrical properties by tuning the frequency and strength of the electric field. cDEP is a variant of DEP that uses fluid electrode channels which are isolated from the main microfluidic channel and thus enable sterile sorting. For further information on cDEP and its applications, see [29-31].

This experiment was performed using the cDEP system described in [32] (microfluidic device Design 3). Distilled water containing a mixture of two types of polystyrene beads – 1 µm diameter red fluorescent particles and 10 µm diameter green fluorescent particles (Fluoro-Max R0100 and Fluoro-Max G1000, Thermo Scientific) was perfused at a rate of 0.005 mL/hr through the microfluidic device using a syringe pump (PhD Ultra, Harvard Apparatus). The conductivity of the particle solution was measured to be 25 µS/cm (SevenGo Pro Conductivity Meter, Mettler-Toledo). The electric field was applied as a sinusoid with 50 kHz and 400V RMS. Images were acquired at 19 Hz with a color camera (Leica DFC420) mounted on a Leica DMI 6000B microscope equipped with fluorescence illumination and filters and had a size of 648x864 pixels at a resolution of 0.83 µm/pix with 5X magnification. A representative image is shown in Figure 3.

![Figure 3: a) Example frame from the cDEP microfluidics experiment. b) 64x64 pixel close-up of particle field.](image-url)
Based on the acquired images, it was determined that the green particles had an apparent size of about 16 pixels in diameter and an average RGB color vector corresponding to (75,220,255) – almost pure cyan, while the red particles were approximately 2-3 pixels in size and were almost pure red with a color of (255, 68, 55).

RESULTS

Effect of Bayer Filtered Images on PIV

The results of the error analysis of single component velocity tests for uniform flow using three differently colored particles is shown in Figure 4. Both Bayer filtered and full-spectrum RGB images were tested, as well as grayscale-converted Bayer filter images. Native grayscale fields were also examined to provide a baseline for traditional scalar PIV methods for comparison to the new methods.

![Bayer Filtered Images](image)

**Figure 4**: Error analysis as a function of x displacement for color and grayscale images of multispectral particle fields constructed with and without a simulated Bayer filter. RPC is a standard scalar cross correlation, qRPC is a quaternion method, and eRPC is a color ensemble method. All measurements are in pixels. a) bias error in $u$ velocity. b) random error in $u$ velocity. c) bias error in $v$ velocity. d) random error in $v$ velocity.

From the results it was immediately apparent that every processing type when used with image data that had been previously Bayer filtered and demosaiced, whether RGB or traditional PIV on grayscale-converted images, demonstrated severe bias and random error fluctuations as a function of displacement. This effect was similar to classical peak locking (seen to a certain extent in the $u_{bias}$ error in Figure 4a) for the measurements made with native grayscale or full spectrum images, but instead had a period of 2 pixels. This detrimental behavior is attributed to the Bayer filter itself, which also has a period of 2 pixels. Examination of the demosaiced images showed clear checkerboard patterns in the particle image data (see for instance Figure 5), and it is likely that this was the root cause of the periodic errors. Errors were lowest when the displacements were an even multiple of 2, and were highest when the displacement was mismatched with the Bayer filter at a single pixel.
Although the magnitudes of the resultant errors remained reasonable (below about 0.1 pixels/frame), they were significantly higher than the correlations computed on unfiltered images, which had bias errors that remained less than 0.05 pixels/frame even for large displacements (8 pixels in a 32 pixel window resolution), and random errors between about 0.02 and 0.04 pixels/frame. It is probable that more sophisticated post-processing of the demosaiced images could alleviate this aliasing artifact in the images. As it was not the intent of this work to evaluate the performance of different image processing techniques, for the remainder of this work, only color images created without the use of the Bayer filter were analyzed. However, these results highlight an important issue that needs to be addressed if color PIV processing is performed and additional testing with experimental images and commercial software needs to be performed to quantify to what extent this problem affects color images in practice.

Returning to the comparison of the different color image processing algorithms versus a traditional grayscale approach, there appears to be little effect on the bias errors in either the x- or y-directions (Figure 4a and c). However, the random error dropped by about ½ in both directions, falling from about 0.03 pixels/frame to less than 0.02 in x (Figure 4b), and from about 0.03 to about 0.015 pixels/frame in y (Figure 4d). This is consistent with the proposed mechanism that suggested that uncorrelated noise between color channels would be suppressed by preserving the full color information in the cross-correlation. Additionally, the color ensemble correlation performed slightly better than quaternion cross-correlation, though the difference was slight.

Linear unmixing of a color vector cross correlation was not tested on these images since it would have resulted in three individual velocity fields each made on a particle field having 1/3 the seeding density that the other methods were testing, making it an unfair comparison.

Synthetic Image Tests of Velocity Separation

Moving to the results of the two component flow tests, both the quaternion cross-correlation with velocity separation and the color vector cross-correlation with linear unmixing of the correlation plane were tested. Results of the error analysis for each particle type (P1 and P2) for bias and random error in both the x- and y-directions are plotted in Figure 6 for all combinations of x-displacements between 0 and 8 pixels/frame. As previously stated, the y-displacement was fixed at 0.3 pixels/frame for both particle types in all tests.

For these results, only valid measurements (those with an error less than 0.5 pixels/frame) were counted toward the final statistics. Based on the results shown, both methods showed some crosstalk for displacements less than the average particle diameter of 3 pixels. For the quaternion cross-correlation, both particle types showed approximately equal bias errors toward the displacement of the other particle type for displacements less than this threshold, and slightly higher random errors for particle 2. Examination of the correlation planes yielded the obvious result that once the particle displacements were too close together the peaks began to merge into one, distorting the shape of both and biasing the subpixel peak location toward the position of the other peak.

In contrast, the bias errors were lower for particle 1 using the color-vector cross-correlation approach than for particle 2. Additionally, for particle 1 the bias was toward particle 2, while for particle 2 the bias was away from particle 1. The cause for this seems to be that crosstalk in the linear unmixing led to a small amount of the correlation peak energy of particle 2 being deposited in the unmixed correlation planes for particle 1, biasing the result toward particle 2. This process, on the other hand, led to missing energy for particle 2 on the side of the correlation peak towards particle 1, biasing the peak detection algorithm away from the other correlation peak. Although not tested here, the exact amount and behavior of this error due to the linear unmixing is likely dependent on the exact color values chosen for each particle type, and this behavior will not necessarily generalize to any particular choice of particle 1 and particle 2.

Overall the errors for the quaternion cross-correlation (left column, Figure 6) were considerably worse than for the color vector processing (right column, Figure 6). In fact, the errors contours displayed in a) and b) actually understate the magnitude of the true errors that were typical for this method. Comparing Figure 6c) to f), it can be seen that for the color vector cross-correlation method, the valid vector percentage was very near 100% for the conditions tested.

Figure 5: Use of a Bayer filter can result in a checkerboard appearance in images of small particles. a) full spectrum RGB data. b) RGB data after Bayer filter demosaicing. c) full spectrum RGB data converted to grayscale. d) Bayer filtered image after conversion to grayscale.
However, the quaternion cross-correlation failed more than 50% of the time in many cases, and often had nearly 0% valid detection probability. This was due to several factors. The first was the previously mentioned merging of the correlation peaks. Second, the selection of two valid peaks often proved problematic, with the second true signal peak sometimes being confused for noise peaks of similar height. Finally, even if the peaks were correctly selected, the sorting algorithm often had trouble automatically assigning each decomposed subimage to the proper particle type. Even if the last two problems could be addressed, the first is likely insurmountable without significant new development of the quaternion cross correlation theory.

With outliers included in the error analysis, instead of peaking near 0.5 pixels (Figure 6a), bias errors for the quaternion cross correlation method actually rose to around 1-2 pixels/frame in the particle displacement overlap region before returning to values below 1 for greater separations (results not shown). Random errors were even worse, peaking as high as 6 pixels/frame before decreasing to values between 1-3 pixels/frame.
Figure 7 shows a close up of the results for the color vector cross-correlation method for 0 displacement in x for particle 1, and 0 to 8 pixels displacement for particle 2. These conditions correspond closely (albeit with different seeding densities) to those used in generating the uniform flow images for which results are plotted in Figure 4. Therefore, we can compare the results for traditional or color PIV on a field with a single flow component to the results for a flow with two velocity components.

![Graphs showing bias and random error](Image)

Figure 7: a) bias and b) random error on the color vector cross-correlation measurement of the u-velocity of particle 1 and 2 vs. increasing displacement of particle 2 for a fixed particle 1 displacement of $U_{p1}=0$. Errors are higher in the regions that the particle displacement differ by less than the width of the particles. Values are the same as in Figure 6d) and e).

For particle 1, which has 0 velocity in the x-direction, the $u$-bias error shown in Figure 7a) was very close to zero, as could be expected from the single component $v$-bias errors plotted in Figure 4c). For particle 2, although the bias error did climb to about 0.15 for displacements less than one particle width from 0, by 3.5 pixels displacement the trend has returned to a very close match of the errors that could be expected if particle 2 were the only flow tracer in the image. The random errors shown in Figure 7b) are slightly elevated, however, as compared to the single component uniform flow as plotted in Figure 4b). For particle 1, random errors peaked near 0.045 pixels/frame before returning to the baseline value near 0.03, while the errors for particle 2 climbed to between 0.035 and 0.04 and remained there for the range of displacements tested. This was in contrast to the random errors reported for single component flow data processed with full-color algorithms, which showed values around 0.02 pixels/frame, though it compared much more favorably to the grayscale processing which had random errors nearer to 0.03 pixels/frame.

**MicroPIV in cDEP Channel**

To demonstrate the ability of the color vector cross-correlation method to successfully resolve multiple velocity components in real experimental images, the algorithm was tested on microPIV results of an experiment featuring two particle types of different size and color. In addition, as described in the methodology, cDEP was used to exert a horizontal force on the large green particles, sorting them from the red particles which generally behaved as flow tracers as the cDEP force was negligible due to their smaller volume. The geometry of the full microchannel included a number of symmetric and asymmetric diverging regions. A field of view near the end of the cDEP region featuring a symmetric divergence region was selected for analysis. At this point, the amount of separation of the larger particle phase was near the maximum achieved for this geometry and field settings. Since the quaternion cross-correlation paired with velocity separation proved to perform so poorly with synthetic images, results for this method on these images are not presented here.

Based on manual analysis of the data, it was determined that the smaller red particles had an intensity profile that could be approximated by the color vector $(255, 68, 55)$ and the larger “green” particles (which appeared in these images to be closer to cyan) had a color corresponding to the vector $(75,220,255)$. These color vectors were then used to define the response functions needed to linearly unmix the correlation plane and identify each velocity component.

Images were processed using a Gaussian image windowing with a resolution of 32x32 pixels on a 32 pixel grid (0% overlap), and correlated using a single pass of a RPC kernel. No validation (other than the peak ratio filtering discussed below) was performed on the resulting vector field. Instantaneous results were favorable, but due to the low seeding density for the green particles, especially on the right side of the image, ensemble correlation of the three-color correlation plane over the sequence of images was used to capture the average velocity over a larger region. Additionally, since in some areas near the right wall of the channel no green particles ever appeared, the peak ratio was used to filter out these regions. Based on examination of the results, correlations with a peak ratio less than 12 were blanked out of the displayed vector plots in Figure 8a.
Figure 8: a) Time-averaged ensemble correlation of the experimental images using the color vector method with linear unmixing. b) Slip velocity between the larger and slower green particles and the red particles. The vector scale is the same for both plots.

As can be seen in the figure and the original video, the larger green particles largely settled to the bottom of the microchannel and had a much lower transit velocity (around 2 pixels/frame at the beginning of the divergence) than the average speed of the red particles (above 4 pixels/frame at the same location), which were better flow tracers and stayed suspended throughout the depth of the fluid. Although the cDEP procedure had a significant effect on the particle position over the length of the entire microchannel, as evidenced by the observed separation by type, due to the small magnitude of the forces involved it was not possible to isolate a slip velocity for the green particles transverse to the mean direction of the smaller red particles (Figure 8b). However, the apparent slip velocity between the two phases was readily apparent, and only in the region near the right wall did there appear to be significant contamination of the velocity signal for green particle from the red. This was because very few green particles reached this region whereas the red particles were densely and uniformly throughout the flow, so it only took a small amount of leakage between the unmixed correlation planes for the signal from the red to overpower the small signal from the green. This can be seen in the patchy areas of disorganized and low velocity flow near the wall and upper right of the image.
More concerning, however, was the blocky appearance of the slip velocity vector field (and individual vectors, though it was harder to spot there). This appeared to have the typical characteristics of peak locking. Examination of the raw $x$- and $y$-displacement fields for the red (P1) and green (P2) fields showed this much more clearly. Figure 9 contours these values for the red particle field; results for the green particle field were similar. Qualitatively, the measured displacements appeared to be grouped around even integer displacements, with sharp transitions between regions of different velocities. Plotting a histogram of all measurements (Figure 10) helped to quantify the severity of the problem. For displacements of both particle types in either $x$ or $y$, there were almost no values near odd integers, and very large peaks for even values. Although peak locking is a common problem for PIV cross-correlation, and ensemble correlation of the correlation planes in time tends to exacerbate the error, it typically has a period of one integer pixel, not two. This suggests that source of the problem was not traditional sources of PIV correlation error, but rather the error introduced by the camera’s demosaicing software, as was previously suggested by the results of synthetic image testing (Figure 4).

![Figure 9: Raw particle displacements showed a clear bias toward even integer displacements. a) Horizontal displacements for the red particle field. b) Vertical displacements for the red particles.](image)

![Figure 10: Histogram of the measured displacements in the x- and y-direction for all vectors measured from both particle types. There is a clear periodic pattern with peaks near even integer values.](image)

Despite the shortcomings introduced by the Bayer filter, it is clear from these results that both velocity components could be resolved, even under challenging microPIV conditions. Additional work on image dealiasing and filtering may provide a solution to the peak locking problem created by the Bayer filter.

**DISCUSSION AND CONCLUSIONS**

Synthetic image tests on flows with only a single velocity component demonstrated a clear advantage for color correlation methods over traditional single channel images. For the two color correlation methods tested (quaternion cross correlation, and color ensemble cross correlation) the bias errors were similar to a traditional grayscale image processing, but the random errors were reduced by almost a factor of one half (Figure 4), with the color ensemble method being slightly better than the quaternion method. From examination of the correlation planes and images, it is hypothesized that the reason for this improvement is that noise tends to be decorrelated between color channels, while
the displacement signal is the same. As such, the signal to noise ratio of the resulting correlation plane is increased when multispectral image data is used to find the displacements.

However, this was only for images that were generated without the use of a Bayer filter to reconstruct the full color spectrum. The use of this 2x2 color mosaic pattern introduced clear periodic patterns versus displacement with a period of 2 pixels in both the bias and random errors. Errors were increased the magnitude by 2 to 3x or more versus images without this problem. These results send a clear warning sign to any researcher using a color camera in their work, because even software conversion of color images back to grayscale is not sufficient to eliminate this problem.

One potential limitation of this work is that only the built-in MATLAB function for demosaicing images was used in conjunction with our synthetic images, and no additional de-aliasing or filtering steps were performed. These filtering steps are nearly ubiquitous in commercial camera software and hardware, though they often trade spatial resolution to eliminate the so-called color moiré or aliasing artifacts seen here. One potential problem with the filtering schemes that might have been selected by manufacturers is that they were likely picked with the goal of increasing the aesthetic appeal of the final results, rather than ensuring that no 2x2 patterned noise remains in the image. However, it is difficult to make universal judgments on whether any particular camera will suffer from these problems since the exact schemes used are typically proprietary and are not disclosed to the public. This means that the user of any given camera would be wise to evaluate the severity of this error under the exact acquisition settings that will be used in their experiment before depending on the images generated for sensitive research needs. Partial validation for the possibility of this error occurring in practice was demonstrated here in the microPIV experiment performed. For the images acquired using this camera (Leica DFC420), cross correlation revealed a dramatic bias error that severely contaminated the resulting vector fields (Figure 8 and Figure 10). A more careful study under more controlled conditions and several common high-speed color video cameras should be undertaken to determine if this is a problem that can be anticipated in normal lab conditions.

One potential workaround would be to use an algorithm that used the entire 2x2 pixel Bayer filter to reconstruct a single pixel in the resulting image. This would effectively downsample the final image by a factor of 2 in each direction as compared to the raw sensor size, but should be effective if implemented properly since each resulting pixel would have been reconstructed in the same way. However, simply subsampling already reconstructed images may not be sufficient to eliminate this error, since the 2x2 checkerboard pattern may have been spread by additional post-processing steps performed by the camera. Further research needs to be done to investigate this problem.

Another alternative would be to use a camera that uses multiple sensors (3CCD or 3MOS devices) or a single multi-color sensor (like the Foveon X3) to record raw intensity data for each of the three color channels at every spatial location. This should completely avoid this error, since no Bayer filter or demosaic operation is required. However, these systems typically start with smaller sensors sizes than equivalent single-sensor cameras, meaning that a trade-off of spatial resolution must be made, which might eliminate any benefit in contrast to the previously described workaround. Additionally, it is not clear whether any high speed or PIV cameras are currently available commercially that feature these systems, meaning that research might be limited to larger interframe times and slower frame rates.

Beyond the potential for enhancing the measurement quality in signal component vector fields, the use of more than one flow tracer to sample multiple types of flow behavior simultaneously is also possible using the methods described here. Such situations are common in multiphase flows, and the results shown prove that it is possible to separate the independent behavior of each particle type with a minimum of cross talk between signals and accuracy and precision similar to that achievable measuring each component of the flow singly. Although quaternion cross correlation paired with velocity separation performed poorly because of the inability of the quaternion cross-correlation to differentiate closely spaced displacements, color vector correlation did not suffer from the same problems, and a linear unmixing approach adapted from the microscopy and satellite imaging communities proved very successful in resolving two simultaneous flow components (Figure 6). Application of the method to experimentally acquired microPIV images showed that the method could successfully detect the velocity signal from two different particle types in real images using both instantaneous and time-ensemble correlations (Figure 8). However, as previously discussed, severe peak locking was observed in the time-ensemble correlation data, likely as a result of the use of a Bayer filter by the camera as seen in the synthetic data tests. This compromises the ability of the method to make accurate measurements of velocity, even though the displacement peaks are readily detectable.

In conclusion, this paper introduced and explored three correlation methods for employing color data in PIV processing. It was shown that for single component flow data seeded with particles of multiple colors, multispectral image data could be used to reduce the random error of PIV methods by about a factor of two as compared to grayscale processing of the same images. Additionally, two methods adopted from the signal and image processing communities, linear unmixing and velocity phase separation, were tested to separate out the individual signals from two flow tracers of differing velocities. Although quaternion cross-correlation paired with velocity phase separation failed due to an
inability to distinguish displacements separated by less than a single particle diameter, color vector correlation avoided this problem, and linear unmixing of the resulting correlation planes yielded results similar to measuring each flow component independently. Finally, it was shown that influence of the Bayer filter caused a severe peak-locking error on PIV fields measured using color data for both synthetic and experimental images. Whether this error is common to all commercially available color camera systems is an issue that needs to be addressed before color processing in PIV becomes routinely viable.

ACKNOWLEDGMENTS
The authors would like to thank Jaka Cemazar for his help performing cDEP experiments.

REFERENCES


