Image cleaning

Cimage_mask,image,info
Step 1. Average all the usable masked pixels in each column including buffer and dark for the 18 binned lines for bin 1, 9 lines for bin 2 and 4 lines for bin 4. Floating point array of 1052 columns for bin 1, 540 columns for bin 2 and 284 columns for bin 4.
Step 2. Subtract this line from every line in the image.

Cimage_dark, image, info
The dark region has had subtracted from it the dark current generated in the mask area by the above procedure. It should be a nominal zero value except for drift and temperature change during imaging
Step 1. Average the last twelve columns of each line and produce a floating point array of n lines long.
Step 2. Apply an 11 point smoothing to this dark array. [I have tried a number of different smoothing approaches including using median and averaging of different lengths. For the recent dark images the 11 point smoothing produced the minimum observable degradation.]
Step 3. Subtract the smoothed value from each line in the array.

The actual code used is given below.

What are the residual errors from this method?

1. The masked area pixels are closer to the hot amplifier than the image pixels. The temperature of the CCD will rise during imaging so insufficient dark current may not be removed.
2. There is an unexplained negative value in the binned area prior to the dark area. This is the biggest error. For bin 2 it is about 15-30 DN.
3. Other?

Essential algorithm extracted from code below

```
 ave_lines,image,21,20+maskbinend,masked          ;All columns in masked
for I = 0L,nlines-1 do a(0,i)=image(*,i)-masked       ;all columns have masked subtracted
ave_cols,image,nc-12,nc-1,dark
dark   = smooth(dark,11)
for i  = 0L,nlines-1 do a(12:(540-17),i)  = image(12:(540-17),i)-dark(i)*bin          ;Note bin not bin^2
for i  = 0L,nlines-1 do a(0:12,i)  = image(0:12,i)-dark(i)
for i  = 0L,nlines-1 do a(nc-16:nc-1,i)  = image(nc-16:nc-1,i)-dark(i)
```

```
;clean by alan 29 sep 06;
function clean,z,info
   zmask  = cimage_mask(z,info) ; clean image using 18 masked ccd rows
   zmask_dark  = cimage_dark(zmask,info)    ;subtract mean of 12 dark pixels per line
       return,zmask_dark
       end
```

```
; function cimage_mask.pro                        verison 1 / 30 Nov 05
;
;created: 30 nov 05, Alan Delamere
;
```

```
;Takes an image and subtracts the average of the masked lines from each line.

function cimage_mask, image, info
sz  = size(image)
a = image
bin  = info.bin
   case info.bin of
       1: maskbinend  = 20-2 ;bin 1 uses 18 of 20 masked lines
       2: maskbinend  = 10-1 ;bin 2 uses exactly the same area under mask 9 of 10
       3: maskbinend  = 6       ;bin 3 uses unbinned lines 3 to 17
       4: maskbinend  = 4       ;bin 4 uses unbinned line 4 to 19
       8: maskbinend  = 2        ;not used see below
       16: maskbinend  = 1
       endcase
nc  = sz(1) & nlines  = sz(2)
if bin lt 8 then begin                          ; trap bin 8&16 cases
  ave_lines,image,20+1,20+maskbinend,masked,m_noise
endif else begin
  if bin eq 8  then masked = (image[*,20+1]+image[*,20+2])/2
  if bin eq 16 then masked =  image[*,20+1]
  m_noise = masked * 0
endelse
for i  = 0l,nlines-1 do a(0,i)  = image(*,i)-masked   ;fixed pattern subtraction
return,a
end

; ------------------------------------------------------------------------
; function cimage_dark.pro                              version 1 / 30 N0V 2005
;
;Created: 30 Nov 2005, Alan Delamere & Patrick Russell
;
;Takes an image and subtracts the average of the dark columns from each line.

function cimage_dark, image,info
sz  = size(image)
a   =  image
bin  = info.bin
nc  = sz(1) & nlines  = sz(2)
ave_cols,image,nc-12,nc-1,dark,m_dark
;dark  = median(dark,11)
dark  = smooth(dark,11)
for i  = 0l,nlines-1 do a(12:(540-17),i)  = image(12:(540-17),i)-dark(i)*bin ;average dark subtraction
for i  = 0l,nlines-1 do a(0:12,i)  = image(0:12,i)-dark(i)        ;average dark subtraction
for i  = 0l,nlines-1 do a(nc-16:nc-1,i)  = image(nc-16:nc-1,i)-dark(i)          ;average dark subtraction
return,a
end
```