

ALMA MATER STUDIORUM  
UNIVERSITA' DI BOLOGNA

SCUOLA DI INGEGNERIA E ARCHITETTURA  
Sede di Forlì

Corso di Laurea in  
INGEGNERIA AEROSPAZIALE  
Classe L-9

ELABORATO FINALE DI LAUREA  
in  
*Fondamenti di Meccanica Orbitale*

**Analisi degli errori tipici dei sensori CCD e CMOS  
nella simulazione di osservabili ottiche**

**CANDIDATO**  
Francesco Monari

**RELATORE**  
Chiar.mo Prof. Paolo Tortora

**CORRELATORE**  
Francesco Fiori

Anno Accademico 2019/2020



You cannot pay attention to what is behind you,  
you got to pay attention to what is ahead of you.

*Michael Collins*



## SOMMARIO

I sensori fotoelettrici di formato CCD e CMOS destinati all'utilizzo in missioni spaziali permettono di catturare immagini dell'ambiente circostante per scopi scientifici o di navigazione: tuttavia, esistono diverse tipologie di rumori che possono degradarne la qualità, tale per cui esse risultano essere molto differenti rispetto al caso di immagini analoghe ottenute tramite simulazione.

Lo scopo del seguente studio è quello di generare osservabili ottiche realistiche, ovvero affette dai rumori caratteristici dei sensori adottati in missioni spaziali a partire da immagini digitali simulate prive di rumore.

Dopo un primo richiamo all'architettura di camera, al funzionamento dei CCD e CMOS e alle loro proprietà, è possibile procedere ad uno studio più approfondito degli errori di cui sono principalmente interessati. Questi sono rumori che si verificano durante la generazione di fotoelettroni, durante la lettura dell'immagine o all'interno dei pixel stessi.

La simulazione degli errori viene effettuata a partire da dati relativi a recenti missioni spaziali, in modo da conferire realistica al lavoro svolto.

Questi rumori simulati sono poi, dove possibile, confrontati con osservabili generate da software terzi, al fine di validare la bontà del codice elaborato.

# INDICE

SOMMARIO .....	5
INDICE .....	6
INDICE DELLE FIGURE .....	7
1. INTRODUZIONE .....	9
2. ARCHITETTURA DI CAMERA .....	11
1.1 TIPOLOGIE DI SENSORI D'IMMAGINE.....	13
1.2 CCD .....	14
1.3 PRINCIPIO DI FUNZIONAMENTO DEL SENSORE CCD .....	15
1.4 CMOS.....	18
1.5 PRINCIPIO DI FUNZIONAMENTO DEL SENSORE CMOS .....	18
1.6 EFFICIENZA DI CONVERSIONE DEI FOTONI .....	19
3. DESCRIZIONE E SIMULAZIONE DEI RUMORI .....	21
3.1 SHOT NOISE DI DARK CURRENT .....	22
3.1.1 GENERAZIONE DI DARK CURRENT .....	22
3.1.2 SIMULAZIONE DEL RUMORE DI DARK CURRENT ED ANALISI DEI RISULTATI OTTENUTI 25	
3.2 BLOOMING .....	27
3.2.1 SIMULAZIONE DELL'EFFETTO DI BLOOMING .....	29
3.3 FIXED PATTERN NOISE.....	33
3.3.1 SIMULAZIONE DEL FIXED PATTERN NOISE ED ANALISI DEI RISULTATI .....	35
3.4 READOUT SMEAR NOISE .....	38
3.4.1 SIMULAZIONE DEL RUMORE DI READOUT SMEAR ED ANALISI DEI RISULTATI OTTENUTI.....	41
3.5 ADC QUANTIZATION NOISE .....	44
3.5.1 ANALISI SULL'INTENSITA' DEL RUMORE DI ADC .....	46
4. CONCLUSIONI .....	53
BIBLIOGRAFIA.....	54
SITOGRAFIA .....	55
APPENDICE .....	56

## INDICE DELLE FIGURE

Figura 1 Raggi di luce attraverso un sistema a singola lente sottile [5] .....	12
Figura 2 Modello di camera con Frontal Image Plane [5] .....	12
Figura 3 Architettura di un CCD [10] .....	14
Figura 4 Trasferimento di carica nel pozzo potenziale [10] .....	15
Figura 5 Raffigurazione dei pozzi potenziali sovrapposti [2] .....	16
Figura 6 Descrizione del passaggio di carica tra due pozzi potenziali sovrapposti [2] .....	17
Figura 7 Schema di lettura per CCD Full Frame [1] .....	17
Figura 8 Schema di funzionamento di un pixel CMOS [10] .....	19
Figura 9 Immagine dell'asteroide Itokawa priva di rumori ottenuta tramite il simulatore PANGU .	21
Figura 10 Principali contributi di Dark Current [9] .....	23
Figura 11 a) Immagine non affetta da rumore generata tramite PANGU b) Immagine contenente lo Shot Noise di Dark Current simulato in MATLAB, a partire dall'immagine senza rumore .....	26
Figura 12 Immagine con Dark Current Shot Noise simulata tramite PANGU .....	26
Figura 13 Blooming di tre regioni contenenti pixel saturi. La direzione verticale è quella di readout del sensore CCD [1] .....	28
Figura 14 Blooming in un'immagine scattata da MESSENGER [5] .....	28
Figura 15 Antiblooming Drain [2] .....	29
Figura 16 Gaussiana di $\sigma=50$ applicata ad una matrice nera 500x500 .....	30
Figura 17 Immagine di Itokawa con tempo di esposizione maggiore generata con PANGU .....	31
Figura 18 a) Maschera contenente solo l'effetto di blooming b) Immagine originale alla quale è stata aggiunta la maschera di blooming (MATLAB) .....	31
Figura 19 Effetto di blooming simulato tramite PANGU .....	32
Figura 20 Dettaglio della maschera contenente anche l'effetto di blooming distribuito b) Immagine affetta da blooming verticale e distribuito. Entrambe le immagini sono ottenute dal codice MATLAB .....	33
Figura 21 Immagine affetta da Fixed Pattern Noise generata con MATLAB .....	36
Figura 22 Immagine ottenuta dopo la calibrazione Flat Fielding (MATLAB) .....	36
Figura 23 Immagine affetta da Fixed Pattern Noise simulata con PANGU .....	37
Figura 24 Dettaglio della Figura 23 .....	38
Figura 25 Dettaglio dell'immagine con Fixed Pattern Noise ottenuta tramite MATLAB .....	38
Figura 26 Funzionamento del Frame Transfer Imager [2] .....	40
Figura 27 Readout Smear Noise, immagine ottenuta durante la simulazione MATLAB tramite applicazione del rumore ad un'immagine pulita .....	40
Figura 28 Maschera d'errore ed immagine finale nel caso reale ( $k=1$ ) .....	42
Figura 29 Maschera d'errore ed immagine finale ( $k=27$ ) .....	42
Figura 30 Maschera d'errore ed immagine finale ( $k=40$ ) .....	43
Figura 31 Effetto di Readout Smear Noise simulato con PANGU .....	43
Figura 32 Output digitale di un segnale analogico [2] .....	44
Figura 33 Matrice <i>Aquant</i> , con gradiente che si sviluppa in direzione orizzontale, quantizzato a 8 bit .....	47
Figura 34 a) DN segnale iniziale ( <i>Aanalog</i> ) b) Segnale quantizzato a 8 bit riconvertito in DN ( <i>AquantDN</i> , valori della sonda MESSENGER) .....	48
Figura 35 Andamento dell'errore di quantizzazione ( <i>Error</i> ) espresso in DN per i valori di una riga (valori tipici della sonda MESSENGER) .....	49
Figura 36 a) Segnale ad 8 bit ( <i>Aanalog</i> ) b) Segnale quantizzato a 4 bit ( <i>AquantDN</i> ) .....	50

Figura 37 a) Segnale iniziale di una riga, espresso in DN ( <i>Aanalog</i> ) b) Segnale quantizzato a 4 bit espresso in DN ( <i>AquantDN</i> ) .....	50
Figura 38 Errore di quantizzazione nel caso a 4 bit con valore massimo normalizzato al valore di bianco (255) .....	51
Figura 39 Errore di quantizzazione nel caso a 4 bit per una riga, espresso in DN.....	51



# 1. INTRODUZIONE

Questo elaborato presenta una analisi e studio degli errori tipici che possono insorgere durante l'acquisizione di un'immagine per mezzo di un sistema di camere ottiche e, più precisamente, a seguito dell'utilizzo dei sensori fotoelettrici di tipologia CCD e CMOS.

L'obiettivo finale consiste nell'applicazione di appositi algoritmi per simulare l'effetto delle più comuni e rilevanti forme di errore introdotti dall'elettronica, al fine di creare osservabili ottiche di missioni spaziali il più possibile realistiche a partire da immagini simulate prive di rumore.

Il lavoro qui di seguito sviluppato è cominciato durante il tirocinio curriculare svolto presso il Tecnopolo di Forlì, durante il quale si è svolta la parte preliminare di ricerca sulle possibili fonti di rumore di cui queste tipologie di sensori digitali sono tipicamente interessate.

Dopo una introduzione atta a descrivere una possibile architettura di camera, si procederà con un'analisi sul funzionamento dei CCD e dei CMOS, per descriverne in maniera dettagliata le caratteristiche rilevanti ai fini dello studio.

Successivamente si concentrerà l'attenzione sulle fonti di rumore che possono influenzare la corretta rappresentazione delle immagini: si tratteranno quindi errori dovuti alla generazione involontaria di fotoelettroni, come per il White Noise e per lo Shot Noise dovuto alla Dark Current, ma anche possibili rumori che si verificano durante la fase di lettura dell'immagine come il Readout Smear Noise.

Nella fase di creazione di una immagine il più possibile priva di errori, è fondamentale assicurare il perfetto funzionamento dei pixel all'interno dell'array del sensore, ma questo non sempre si verifica: possono infatti insorgere degli effetti indesiderati, quali il Blooming o il Fixed Pattern Noise causati, rispettivamente, da un accumulo di fotoelettroni maggiore del valore soglia di saturazione del pozzo potenziale (facendo comparire tipiche linee di pixel saturi nell'immagine) e da una non uniformità riguardante la sensibilità di ogni singolo pixel.

A conclusione dell'analisi teorica, sarà inoltre introdotto l'effetto causato dal rumore di conversione del segnale raccolto (sotto forma di numero di elettroni) in un segnale digitale (un valore di tensione, opportunamente convertito nel cosiddetto Data Number, ovvero il valore grezzo di lettura del pixel), passaggio necessario per la creazione di un'immagine finale e ottenuto tramite l'utilizzo dell'Analog-to-Digital Converter (ADC).

L'analisi delle tipologie di errori descritte ha posto le basi per la creazione di funzioni e codici per mezzo del programma MATLAB®, in modo tale da ottenere una visualizzazione grafica degli effetti negativi dei rumori descritti durante la trattazione teorica.

Utilizzando un software per la generazione di osservabili ottiche fotorealistiche (PANGU, Planet and Asteroid Natural Scene Generation Utility), è stato possibile creare dapprima immagini senza l'aggiunta di rumori, in seguito le stesse immagini con rumore.

Esse sono state utilizzate per ottenere immagini di confronto per l'output delle simulazioni di rumore ottenuto dal codice sviluppato per questo elaborato.

Come modello per la generazione delle immagini è stato scelto l'asteroide areosecante 25143 Itokawa, dal quale è stata ottenuta un'immagine simulata priva di errori alla quale saranno applicati singolarmente tutti gli effetti dei rumori studiati.

## 2. ARCHITETTURA DI CAMERA

Le varie tipologie di camere ottiche utilizzate a bordo degli spacecraft possono differire in funzione delle singole missioni da soddisfare, ma presentano una medesima architettura di base che vede l'utilizzo di strumenti ottici, quali lenti e specchi, e l'impiego di sensori rilevatori.

Uno dei parametri fondamentali che caratterizzano le camere ottiche è rappresentato dalla lunghezza focale  $f$  di una lente. Essa può essere espressa, in prima approssimazione, dalla formula delle lenti sottili (2.1), dove è possibile notare che  $f$  è funzione della distanza tra il target e la lente  $s_1$ , quella tra la lente e il sensore  $s_2$ , l'indice di rifrazione  $n$  ed i raggi di curvatura della lente sia dal lato esposto all'oggetto fisico, sia dal lato del sensore (rispettivamente  $R_1$  e  $R_2$ ). [5]

$$\frac{1}{s_1} + \frac{1}{s_2} = (n - 1) \left( \frac{1}{R_1} - \frac{1}{R_2} \right) = \frac{1}{f} \quad (2.1)$$

Considerando un fascio di raggi di luce paralleli fra loro e perpendicolari al piano comprendente la lente sottile, la lunghezza focale  $f$  rappresenta la distanza fra il punto di convergenza di tali raggi dopo essere stati deviati nel passaggio attraverso la lente stessa e il piano che la comprende.

Questo punto è detto punto focale, e facendo coincidere la lunghezza focale con  $s_2$ , giustifichiamo la validità del modello semplificato a singola lente sottile appena descritto.

Tale architettura può essere anche descritta come un sistema ulteriormente semplificato di tipo *pinhole*, ovvero un sistema composto da un foro in una superficie piana, che separa l'oggetto dal piano dell'immagine.

In questo caso, l'unico effetto apprezzabile risulta essere un capovolgimento dell'immagine in ingresso, analogamente al caso della lente sottile (escludendo altri effetti, quali profondità di campo illimitata, aumento dell'effetto dovuto a fenomeni di rifrazione).

Nel caso del pinhole, la lunghezza focale viene definita come la sua distanza da una superficie di riferimento, dove è collocato il sensore d'immagine.

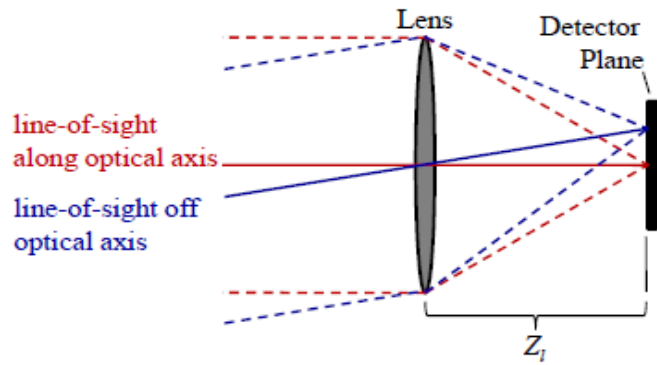


Figura 1 Raggi di luce attraverso un sistema a singola lente sottile [5]

Il modello pinhole ha quindi un vantaggio nella semplificazione della geometria del sistema, mettendo in relazione un punto preso sul piano del sensore con il corrispondente vettore della linea di vista  $(e_i)_c$  tramite:

$$(e_i)_c = \frac{1}{\sqrt{X_i^2 + Y_i^2 + Z_i^2}} \begin{bmatrix} -X_i \\ -Y_i \\ Z_i \end{bmatrix} \quad (2.2)$$

dove  $X_i$  e  $Y_i$  sono le coordinate del piano  $X - Y$  in cui il raggio proveniente dalla  $i$ -esima sorgente lo intercetta, mentre la distanza tra sensore e lente, sempre nello stesso piano, è data da  $Z_i$ , come si nota dalla Figura 1. [5]

Può inoltre risultare vantaggioso aggiungere a questa geometria il *frontal image plane*, ovvero un piano immaginario parallelo al sensore posto di fronte alla lente: in tal modo è possibile operare con un'immagine finale non più ribaltata, ma orientata come ciò che vedrebbe un occhio umano collocato nel punto focale.

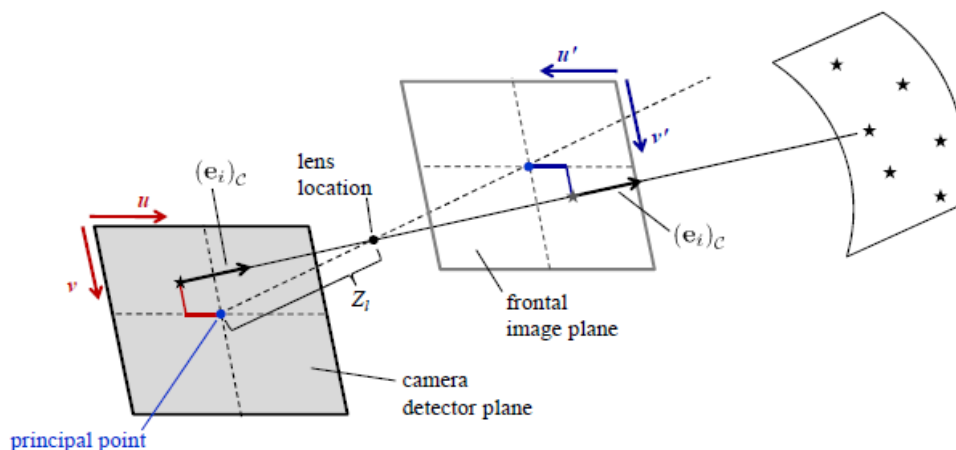


Figura 2 Modello di camera con Frontal Image Plane [5]

Per la descrizione del sistema di riferimento di questo piano frontale è possibile utilizzare il sistema di coordinate  $[u', v']$ , la cui origine è collocata nell'angolo in alto a sinistra del piano, se osservato dal punto focale.

Queste coordinate  $[u', v']$  sono legate alle coordinate  $[u, v]$  del sensore tramite la relazione (2.3) in cui si indica con  $v_{max}$  la coordinata dei pixel nell'ultima riga del sensore. [5]

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & v_{max} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (2.3)$$

Pertanto, il *frontal image plane* introdotto in precedenza può essere utilizzato come piano in cui collocare il sensore di immagine.

Per quanto queste semplificazioni rappresentino un'analisi coerente con i sistemi utilizzati in ambito spaziale, è altrettanto evidente che le camere realmente impiegate differiscano da quelle descritte sotto numerosi aspetti: la necessità di ridurre il più possibile le dimensioni del modello di camera prevede infatti un incremento del numero di lenti e specchi, mentre la presenza dei problemi di aberrazione dovuti all'impiego di una lente sferica senza correzioni aggiuntive, giustifica la presenza dei *field correctors* o l'eventuale uso di lenti paraboliche, allontanandosi dall'ipotesi semplificata di lente sottile. [9]

Nei paragrafi successivi segue una descrizione delle varie tipologie di sensore presenti nei sistemi di acquisizione con le loro caratteristiche principali.

## 1.1 TIPOLOGIE DI SENSORI D'IMMAGINE

I sensori che vengono maggiormente impiegati nelle camere ottiche sono di due tipologie, i Charged-Coupled Devices (CCD) ed i Complementary Metal Oxide Semiconductors (CMOS): questi sono sensori fotoelettrici, spazialmente discretizzati in *pixels* (ovvero piccole celle poste sulla superficie del chip). Il funzionamento di entrambi i modelli si basa sulla generazione degli elettroni a partire dai fotoni incidenti, sulla raccolta di queste cariche in aree dedicate e sul loro trasferimento al registro di lettura al fine di una successiva digitalizzazione dell'informazione raccolta.

## 1.2 CCD

La tecnologia CCD venne introdotta agli inizi degli anni '70 ed è caratterizzata da un array di capacitori MOS, i quali sono costituiti da semiconduttori a ossido metallico fotoattivo e disposti secondo uno schema a griglia (Figura 3). [5]

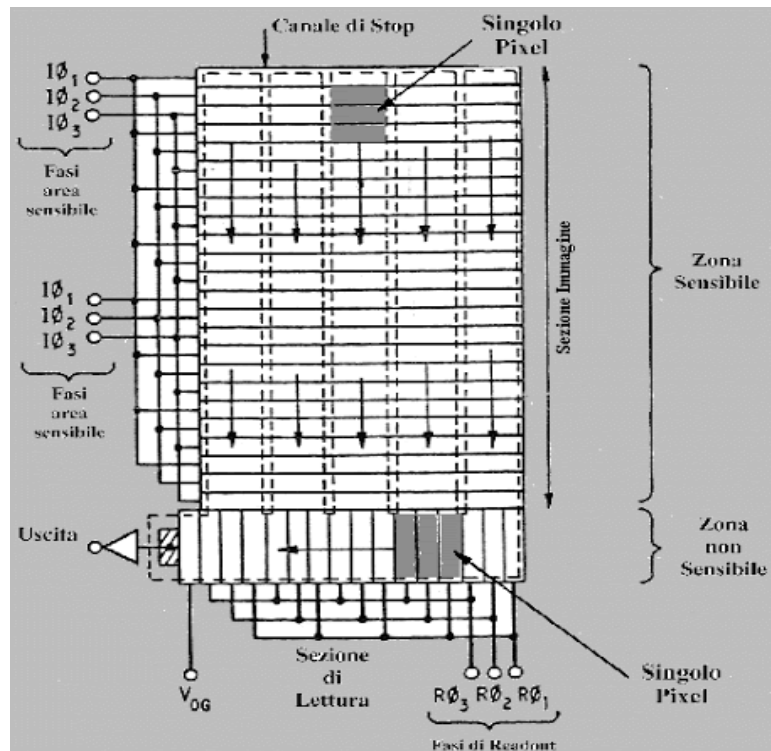


Figura 3 Architettura di un CCD [10]

Le prime architetture di questo sensore prevedevano uno schema di lavoro a pieno formato (chiamato *Full Frame*), dove ogni singola cella aveva sia la funzione di rilevatore fotoelettrico, che di strumento di passaggio di carica, permettendo quindi lo scambio del segnale tra i vari pixel [10].

Tuttavia, uno dei principali svantaggi legati a questa particolare tecnologia è la necessità di comprendere anche un otturatore (o *shutter meccanico*) atto ad evitare diverse fonti di rumore provocate dalla generazione di segnali indesiderati durante la fase finale di lettura del segnale raccolto.

La risoluzione di questo problema ha guidato lo sviluppo della successiva tipologia di CCD, la quale è caratterizzata da una giunzione  $n + p$  per la rilevazione e da un successivo trasferimento di carica dal pixel al registro di lettura completamente autonomo rispetto al rilevatore.

### 1.3 PRINCIPIO DI FUNZIONAMENTO DEL SENSORE CCD

Come prima operazione, il sensore deve raccogliere e trattenere nella sua superficie l'informazione legata ai fotoni incidenti provenienti da una sorgente luminosa esterna alla camera: se questi riescono ad interagire con il semiconduttore del pixel, ovvero possiedono un'energia maggiore o uguale al valore soglia chiamato *band gap* (i.e.  $1.14\text{ eV}$  a  $302\text{ K}$  per il Silicio), allora sono in grado di generare delle coppie *electron-hole* nella superficie del materiale dedicato alla rilevazione.

Le coppie *electron-hole* sono caratterizzate da elettroni liberi (quindi cariche negative) che lasciano al loro posto una lacuna (*hole*, di carica positiva), il cui numero dipende dal numero di fotoni incidenti alla parte sensibile del sensore e dalla loro energia.

Nel caso in cui il fotone non abbia un'energia sufficiente per superare il valore di *band gap* del materiale, questo attraversa il chip, come avviene ad esempio nel caso del Silicio che risulta essere "trasparente" nel range infrarosso. [1]

Come è possibile osservare in Figura 4 gli elettroni generati nel *fotodiodo* (PD) possono raggiungere il pozzo potenziale solo tramite i *Gate di Trasferimento* (TG), i quali sono in grado di creare una barriera di potenziale temporanea grazie all'applicazione di un certo valore di tensione:

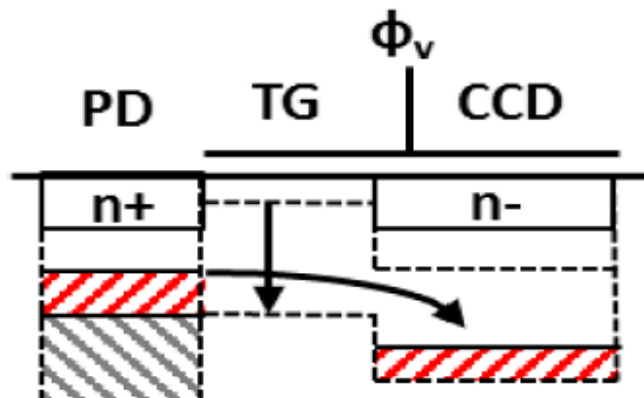


Figura 4 Trasferimento di carica nel pozzo potenziale [10]

se questi sono in modalità "On", i fotoelettroni sono liberi di passare attraverso il gate in direzione del pozzo potenziale (*potential well*) fino a che il valore del potenziale nella regione *n+* diventi tale da eguagliare quello di barriera.

A questo punto il gate passa in modalità "Off", aumentando sensibilmente il valore del potenziale e permettendo l'acquisizione di elettroni in un nuovo ciclo nella regione *n+*.

La carica che ogni pixel è in grado di accumulare nel suo pozzo potenziale segue un andamento lineare rispetto al tempo di esposizione al raggio incidente e all'intensità di quest'ultimo, fino a che non viene raggiunto il valore di saturazione in cui il comportamento non può essere più considerato lineare.

Tale valore di saturazione compare prima che il pozzo potenziale abbia raggiunto la sua massima capienza (circa il 90% della capacità massima), e solitamente il segnale in uscita viene quantizzato fra il valore minimo e il valore di saturazione. [16]

La particolare disposizione dei pixel nello schema a griglia dell'array è fatta in modo tale per cui si verifica una sovrapposizione dei pozzi potenziali appartenenti ai pixel adiacenti. Questa struttura sta alla base del metodo con cui viene elaborata la carica raccolta: applicando infatti ai capacitori MOS un certo valore di tensione, si ottiene lo spostamento degli elettroni da un pozzo potenziale ad un altro (Figura 5).

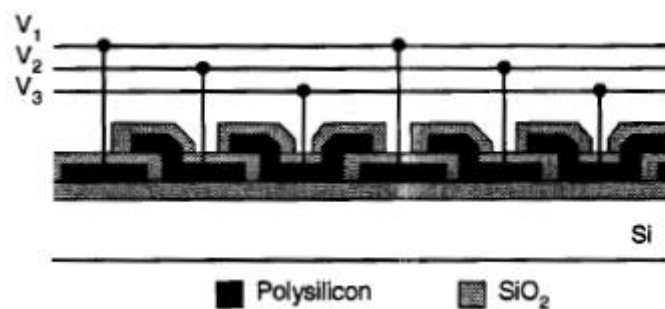


Figura 5 Rappresentazione dei pozzi potenziali sovrapposti [2]



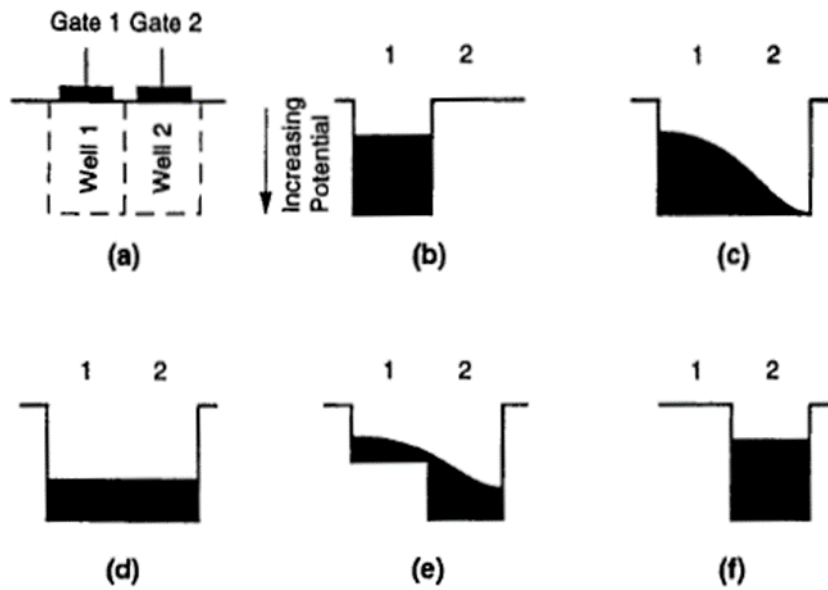


Figura 6 Descrizione del passaggio di carica tra due pozzi potenziali sovrapposti [2]

Nell'architettura Full Frame si procede operando questo trasferimento in serie dalla riga più alta a quella più bassa dell'array (Figura 6): in questo modo quando le cariche contenute nei pixel arrivano in corrispondenza dell'ultima riga, passano nel registro di scorrimento (*serial readout register*).

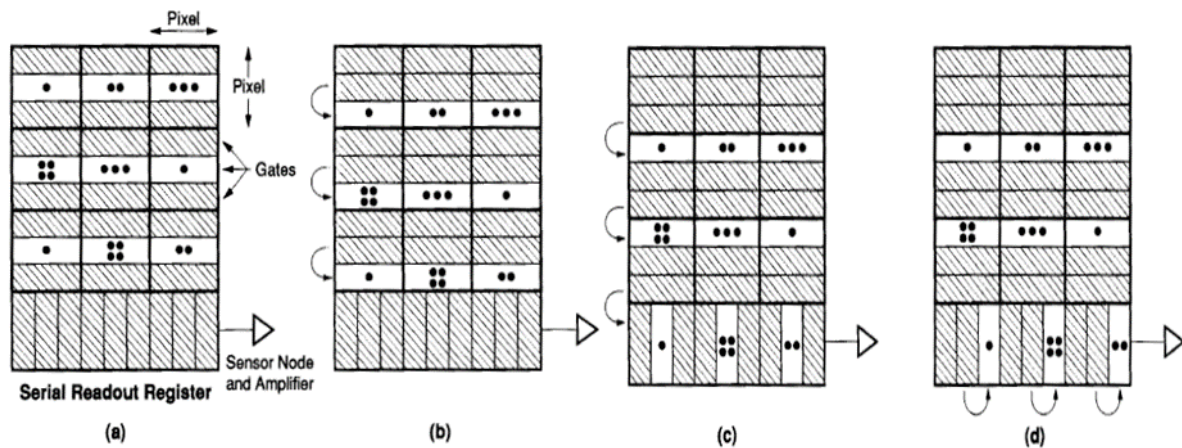


Figura 7 Schema di lettura per CCD Full Frame [1]

Questo output viene proposto in un registro a scorrimento verticale e, una volta registrato, si avanza con la stessa procedura nelle righe sottostanti innescando una logica riga per riga [5].

Affinché si riesca ad ottenere un'immagine digitale finale, ogni pacchetto di carica va mandato dal registro ad un circuito di output (Analog-to Digital converter) in grado di quantizzare il segnale, associando al valore di tensione un determinato *Data Number* (DN).

A causa della sovrapposizione dei pozzi potenziali (un esempio è mostrato in Figura 5) si introduce, tuttavia, una possibile sorgente interna di rumore, il *blooming*, il quale è caratterizzato da spostamenti spontanei di cariche da un pixel saturo verso i pixel adiacenti: per fronteggiare questo problema alcuni modelli di CCD sono dotati di sistemi di *anti-blooming* che riducono il problema a scapito di un abbassamento di sensibilità dell'intero array.

Oltre al blooming, le tipologie di rumore più frequenti sono quelle dettate dal *Readout Smear Noise* (il quale è caratterizzato da un effetto di strisciamento dell'immagine stessa in una direzione preferenziale durante la fase di lettura dell'immagine), dalla componente di *Dark Current* (movimento di elettroni liberi non generati da luce incidente al sensore) e dallo *Shot Noise* (causato anche dalla non uniformità della risposta ai fotoni incidenti di ogni pixel). Altrettanto rilevante è il rumore provocato dalla conversione di un segnale analogico in un dato digitale, ovvero *l'ADC Quantization Noise*.

## 1.4 CMOS

Il formato CMOS dei sensori fotoelettrici rappresenta un'evoluzione del modello CCD per contrastare in modo efficace il problema del blooming d'immagine, senza però alterare significativamente il principio di funzionamento della versione precedente.

Le due architetture risultano simili sotto molteplici aspetti, entrambe fanno uso dei capacitori MOS, ma la principale differenza è rappresentata dal metodo e dalla struttura utilizzati per ottenere la conversione del numero dei pacchetti di carica, ovvero i fotoelettroni generati sulla superficie sensibile, in un valore digitale finale, espresso in *Data Number*.

## 1.5 PRINCIPIO DI FUNZIONAMENTO DEL SENSORE CMOS

La carica ottenuta dal fotorivelatore viene trasportata fino ad una regione denominata *floating diffusion (FD)*, Figura 8, il cui valore del potenziale risulta può essere monitorato attraverso una *source follower (SF)*: entrambe queste componenti si trovano all'interno del pixel e trasmettono l'informazione raccolta al *collettore verticale laterale (COL BUS)*, permettendo quindi la conversione ADC all'interno del singolo pixel. Infatti, non avendo sovrapposizione fra i pozzi potenziali (come accadeva nel caso CCD) è possibile arginare il problema iniziale di blooming.

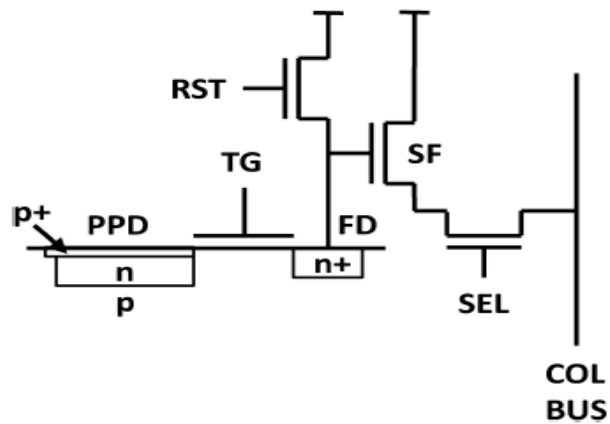


Figura 8 Schema di funzionamento di un pixel CMOS [10]

I CMOS sono diventati una valida alternativa ai CCD a partire dagli anni '90, anche grazie ad un ulteriore vantaggio identificato nel *windowing*, ovvero nella possibilità di riuscire a processare selettivamente porzioni di maggior interesse dell'immagine evitando di analizzare l'intero array di pixel presenti nel rilevatore.

Spostando una parte dell'elettronica nel singolo pixel, è possibile ottenere un'elevata miniaturizzazione e, dunque, i sensori CMOS trovano largo impiego negli strumenti d'immagine impiegati negli spacecraft di nuova generazione destinati alle missioni nello spazio profondo. [11]

Tuttavia, questa tipologia di sensore è caratterizzata anche da effetti svantaggiosi rispetto al classico CCD, quali la necessità di aggiungere un circuito *Analog-to-Digital Converter (ADC Chip)* ad ogni pixel che, oltre a rappresentare una maggiore complessità progettuale, porta con sé un aumento del Fixed Pattern Noise.

Inoltre, materiale presente all'interno della parte fotosensibile del CMOS è silicio in forma maggiormente dopata rispetto a quello presente nel CCD: tale implementazione giustifica una componente di *Dark Current* più significativa e riduce la sensibilità stessa del sensore se utilizzato per esposizioni più prolungate rispetto a quelle nominali di progetto. [1]

## 1.6 EFFICIENZA DI CONVERSIONE DEI FOTONI

Per la descrizione precedente delle strutture dei sensori, è stato considerato un modello di funzionamento riguardante la cattura dei fotoni e la generazione dei fotoelettroni nel singolo pixel ideale.

Infatti, anche ipotizzando che tutti i fotoni emessi dalla sorgente luminosa abbiano un'energia interna maggiore del valore di band gap, nella realtà non tutti riescono effettivamente a produrre fotoelettroni.

Uno dei motivi che impedisce la conversione di tutti i fotoni incidenti in un pixel in fotoelettroni è rappresentato dal *Fill Factor (F)*, il quale indica la percentuale di superficie che effettivamente risulta essere rivestita dai vari strati di materiale fotosensibile utile al funzionamento del sensore.

I fotoni incidenti nella parte restante del pixel (quella non coperta da superficie fotosensibile) non vengono quindi rilevati, riducendo sensibilmente il numero di fotoni effettivamente elaborati.

La *Quantum Efficiency (QE)*, o *efficienza quantica*, riporta invece su che scala di efficienza di conversione si trova il materiale fotosensibile utilizzato, ed è esprimibile come il rapporto tra numero di fotoelettroni e il numero di fotoni incidenti: il termine *QE* indica una media delle efficienze quantiche nell'intera gamma spettrale del sensore, ovvero l'efficienza quantica è funzione della lunghezza d'onda della luce.

Il numero di fotoelettroni effettivamente generati ( $N_{pe}$ ) può essere ricavato tramite l'equazione (2.4):

$$N_{pe} = (QE)(F)(S)A_{pixel}\Delta t \quad (2.4)$$

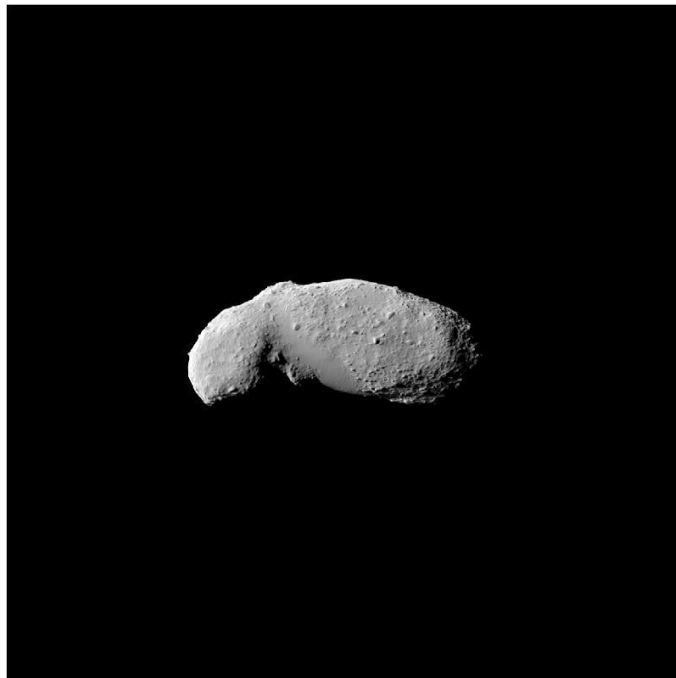
dove la quantum efficiency è legata al fill factor, insieme all'area del pixel ( $A_{pixel}$ ), al tempo di esposizione in cui vengono accumulati i fotoni ( $\Delta t$ ) e al flusso totale di fotoni incidenti  $S$  espresso in  $\left[\frac{\text{fotoni}/\text{m}^2}{\text{secondi}}\right]$ . [5]

### 3. DESCRIZIONE E SIMULAZIONE DEI RUMORI

Di seguito è proposta la trattazione specifica alle sorgenti di rumore che si sviluppano durante l'utilizzo dei sensori CCD e CMOS per la creazione dell'immagine: nel dettaglio si parlerà di Shot Noise di Dark Current, Blooming d'immagine, Fixed Pattern Noise, Readout Smear Noise e ADC Quantization Noise.

Per la maggior parte di questi verrà prima fornito un approfondimento sull'origine di questi rumori e sul loro sviluppo, poi verrà indicato il procedimento seguito durante la ricostruzione pratica degli errori attraverso codice scritto in linguaggio MATLAB® ed il confronto con i risultati ottenuti dal simulatore PANGU (Planet and Asteroid Natural Scene Generation Utility). I codici sviluppati sono stati tutti raccolti nell'appendice.

Ogni tipologia di rumore verrà applicata ad un'immagine priva di errori raffigurante l'asteroide aerosecante 25143 Itokawa, prodotta tramite il simulatore PANGU e mostrata in Figura 9: in questo modo sarà immediato notare gli effetti prodotti dai rumori analizzati.



*Figura 9 Immagine dell'asteroide Itokawa priva di rumori ottenuta tramite il simulatore PANGU*

### 3.1 SHOT NOISE DI DARK CURRENT

Il segnale elettrico di un sensore di formato CCD o CMOS è costituito, per la maggior parte, dalla conversione di fotoni incidenti in fotoelettroni, ma a questo va aggiunta un'altra componente dovuta al moto casuale degli elettroni. Tale moto, causato da effetti termici del materiale, prende il nome di White Noise.

Il white noise (il cui nome deriva dalla presenza di componenti in tutto lo spettro delle frequenze, in analogia alla luce bianca), chiamato anche *thermal noise*, è una tipologia di rumore dovuto al movimento spontaneo di cariche dovuti ad effetti termici del materiale.

Gli elettroni che caratterizzano il segnale del sensore, quindi, sono in grado di muoversi liberamente accelerando o decelerando a seconda della loro energia termica, senza una direzione preferenziale ed in maniera del tutto irregolare.

L'intensità di tale corrente, il White Noise, può essere stimata attraverso semplici equazioni termodinamiche che sono funzione della temperatura assoluta del materiale. [1]

Spesso il white noise è indicato anche come *Floor Noise*, in quanto rappresenta il minimo livello di rumore rilevabile.

Tale valore può essere descritto in termine di deviazione standard di una funzione gaussiana, come in equazione (3.1), la quale considera la capienza massima di ogni pozzo potenziale dei pixel (espressa in numero di elettroni) ed il range dinamico del sensore CCD, ovvero l'intervallo tra minimo e massimo valore assumibile dal segnale generato (indicato nella sua forma lineare oppure nella forma espressa in decibel,  $DR_{dB} = 20 \log_{10} DR$ ). [5]

$$\sigma_{floor} = \frac{N_{max}}{DR} \quad (3.1)$$

#### 3.1.1 GENERAZIONE DI DARK CURRENT

La Dark Current è una corrente che non è legata ai fotoelettroni liberati in seguito all'incidenza di fotoni alla parte fotosensibile del sensore, pertanto rappresenta una forma indesiderata di rumore.

Questo è chiaramente un rumore di natura termica, in quanto composto da un moto indesiderato di elettroni liberi altamente dipendente dalla temperatura operativa del sensore: è dunque possibile catalogare lo Shot Noise di Dark Current come un particolare tipo di White Noise.

All'interno del sensore in formato CCD vi sono numerosi componenti che possono essere considerate responsabili della generazione di questa corrente, come mostra la Figura 10.

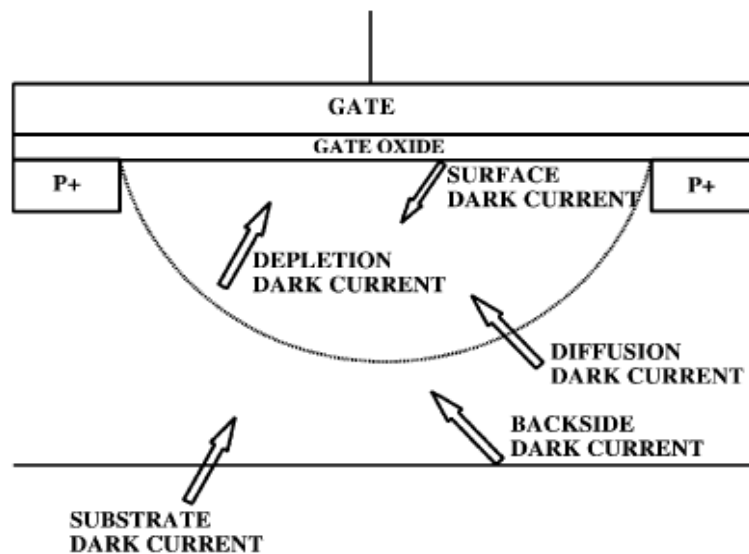


Figura 10 Principali contributi di Dark Current [9]

Il contributo proveniente dalla superficie (surface dark current) rappresenta la componente dominante dell'intera Dark Current.

Esso, tuttavia, non è un valore fisso in quanto varia significativamente tra i vari formati di sensori fotoelettrici: dipende infatti dalle scelte di lavorazione e quindi di crescita della componente di ossido e della passività della superficie. [1]

La Dark Current di diffusione (diffusion dark current) ha invece origine nella zona sottostante al pozzo potenziale, ma questa componente non apporta un contributo rilevante in quanto consiste di elettroni liberi e quindi, a causa del loro moto caotico, provocano una media di corrente uguale a zero per definizione.

Lo stesso accade nel caso della dark current del substrato (substrate dark current), con l'eccezione che quest'ultima segue il parametro fisico della ricombinazione di Auger e, in presenza di silicio dopato almeno quanto il substrato, presenta una sostanziale differenza con la dark current di diffusione per quanto riguarda appunto la distanza media di diffusione. [9]

Infine, si può considerare come ultima componente di dark current quella proveniente dalla parte posteriore del sensore CCD (backside dark current) che, anche se solitamente viene azzerata, può essere fonte d'errore a causa dell'accumulazione di ioni.

Per avere una stima del rumore di dark current, è necessario l'impiego dei dark pixel, ovvero particolari pixel che vengono oscurati dalla luce esterna: il segnale che questi generano, non

essendo causato dai fotoni in ingresso, è pertanto sintomo della sola dark current e quindi può essere utilizzato per stimarne l'intensità (si parla di stima in quanto il segnale ricavato dai pixel è considerato come una media). [15]

Illustrate le possibili fonti di Dark Current, si procede con una quantificazione numerica del numero di elettroni prodotti.

Le equazioni che seguiranno faranno uso di due costanti fondamentali, ovvero la carica dell'elettrone ( $q = -1.6 * 10^{-19} C$ ) e la costante di Boltzmann ( $k = 8.62 * 10^{-5} \frac{eV}{K}$ ).

Il primo dato da calcolare è il valore di band gap energy, esprimibile attraverso l'equazione empirica (3.2) [5] che è solo funzione della temperatura operativa del sensore:

$$E_G = 1.557 - \frac{(7.021 * 10^{-4})T^2}{1108 + T} \quad (3.2)$$

Il valore  $E_G$  viene poi impiegato per il calcolo della costante  $C$ , che prevede anche l'utilizzo, nella sua formulazione, dell'area del pixel  $A_p$ , della temperatura della camera di prova ( $T_{RM} = 300K$ ) e del dato corrispondente alla figura di merito di Dark Current, sempre relativo alla temperatura ambiente,  $D_{FM}$ . [1]

$$C = \frac{D_{FM}A_p}{(qT_{RM}^{1.5})^{-\frac{E_g}{2kT_{RM}}}} \quad (3.3)$$

Giunti a questo punto, è possibile ricavare la densità di elettroni generati da ogni pixel (riferito al tempo di un secondo),  $J_{DARK}$ :

$$J_{DARK} = (CT^{1.5})^{-\frac{E_g}{2kT}} \quad (3.4)$$

L'equazione (3.4) permette, una volta ricavata la Dark Current media prodotta dopo la calibrazione, di determinare la temperatura massima tale per cui l'effetto di questo rumore non risulta più essere trascurabile.

Avendo a disposizione la densità di elettroni di dark current per singolo pixel, si ricaverà il numero totale di elettroni prodotti nel pixel moltiplicando per il tempo d'integrazione  $I_T$ :

$$N_{DC} = \frac{J_{DARK}A_pI_T}{q} \quad (3.5)$$



La Dark Current si differenzia però dal classico White Noise in quanto il suo rumore non segue la classica distribuzione gaussiana, ma la distribuzione di Poisson. Infine, a partire dalla media di elettroni prodotti per pixel, è possibile ricavare il valore del rumore  $\sigma_{DARK}$  [1]:

$$\sigma_{DARK} = \sqrt{N_{DC}} \quad (3.6)$$

### 3.1.2 SIMULAZIONE DEL RUMORE DI DARK CURRENT ED ANALISI DEI RISULTATI OTTENUTI

Si riporta di seguito il procedimento utilizzato durante la simulazione dell'errore di Dark Current attraverso il programma MATLAB®.

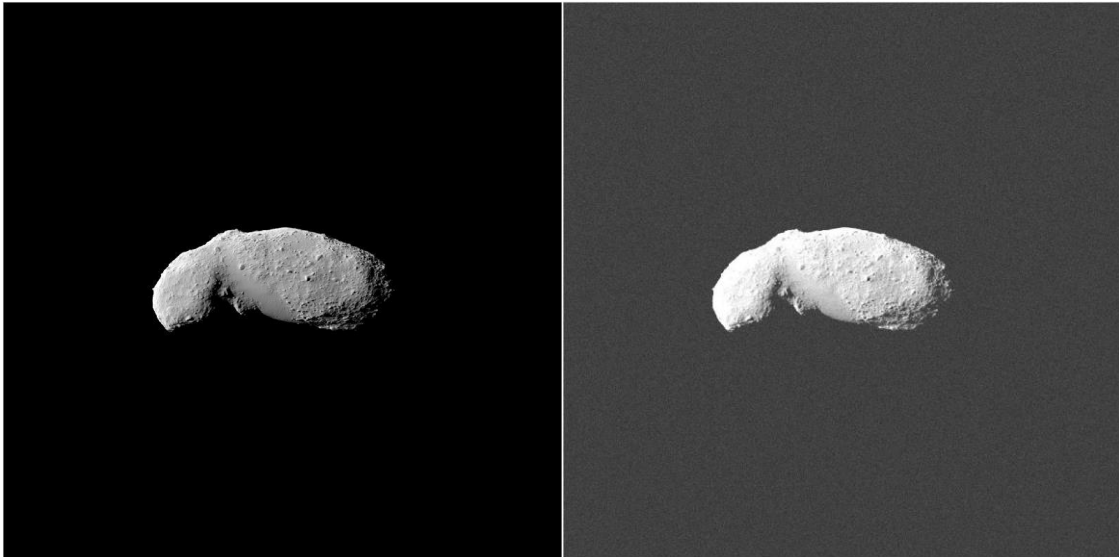
La funzione utilizzata prevede in ingresso alcuni valori riguardanti la struttura del sensore fotoelettrico e alcuni dati specifici dovuti alle sue condizioni di utilizzo: nel nostro caso sono stati considerati i valori relativi al sistema di camera MDIS della missione spaziale americana MESSENGER, per far sì di simulare uno scenario realistico.

I dati richiesti sono la temperatura operativa, l'area di un singolo pixel, il tempo di integrazione ed il valore della figura di merito di Dark Current (nel nostro caso  $T = 243.15 K$ ,  $A_p = 12 \mu m$ ,  $I_t = 0.03s$ ,  $D_{FM} = 0.5$ ). [3][14]

Dopo aver impostato all'interno della funzione il valore della costante di Boltzmann, la carica dell'elettrone e la temperatura ambiente, vengono implementate le equazioni (3.2), (3.3), (3.4), (3.5), (3.6).

Ottenuto quindi il valore di Dark Current Shot Noise, lo si utilizza per creare una matrice  $r$  contenente la sua distribuzione di Poisson attraverso la funzione "poissrnd": questa matrice va applicata poi a quella di partenza (nel nostro caso, una immagine senza rumori rappresentante l'asteroide Itokawa).

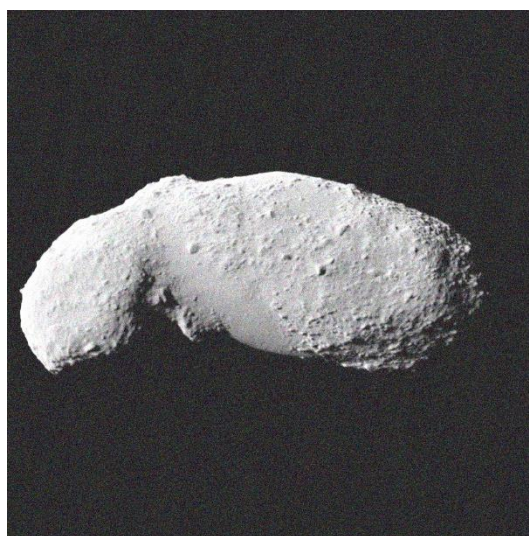
L'effetto prodotto da tale rumore, tuttavia, risulta trascurabile a causa delle favorevoli condizioni operative di MESSENGER, quindi si è scelto di proporre un caso identico, a meno della temperatura operativa, la quale è stata alterata al valore di  $T = 356.15 K$ , in modo da evidenziare chiaramente il rumore generato. (Figura 11.b)



*Figura 11 a) Immagine non affetta da rumore generata tramite PANGU b) Immagine contenente lo Shot Noise di Dark Current simulato in MATLAB, a partire dall'immagine senza rumore*

Viene riportato in Figura 12 anche l'output della simulazione dello Shot Noise di Dark Current ricavata con il software PANGU, utilizzando gli stessi parametri d'ingresso impiegati anche nel codice MATLAB.

Il risultato finale delle due ricostruzioni è molto simile, con l'unica differenza che nell'immagine generata con PANGU il rumore è espresso nei canali rosso, blu e verde, mentre il caso di MATLAB® è rappresentato in scala di grigi.



*Figura 12 Immagine con Dark Current Shot Noise simulata tramite PANGU*

## 3.2 BLOOMING

Come già anticipato nei precedenti capitoli, il blooming è un fenomeno che si verifica durante l'accumulo di fotoelettroni all'interno del pozzo potenziale del pixel ed è un errore strettamente legato alla tipologia di sensori CCD multifase, poiché i CMOS convertono i valori di tensione all'interno di ogni pixel prima della lettura, rendendo il sensore immune a questo tipo di rumore.

Parlare di struttura multifase porta la nostra analisi su un tipo di sensore che prevede la presenza di pozzi potenziali sovrapposti per permettere lo spostamento dei pacchetti di carica al suo interno: per esempio, le architetture Full Frame CCD o Frame Transfer Imager sono entrambe basate sulla struttura appena descritta, e quindi entrambe possono soffrire di questa tipologia di rumore. [8]

Il blooming dell'immagine si verifica non appena il pixel che si sta osservando raggiunge la condizione di "bloomed full well", ovvero quando si ha il raggiungimento della capacità massima del pozzo potenziale: non potendo più accumulare le cariche all'interno del pixel, queste tendono ad espandersi nei pozzi potenziali vicini. [1]

Ciò avviene quando il valore di tensione nel pozzo potenziale del pixel diventa pari a quello della Barrier Phase adiacente, ovvero il canale laterale che separa le colonne di pixel del CCD. [1]

Il movimento indesiderato di questi fotoelettroni è un problema rilevante in quanto va ad alterare la misurazione sotto due aspetti:

- il pixel saturo stesso contiene una carica che non è più proporzionale alla luce incidente, perdendo così l'informazione legata all'intensità massima del soggetto presente nell'immagine;
- i pixel adiacenti ricevono fotoelettroni in eccesso, aumentandone l'intensità del valore registrato.

Nel blooming, la dinamica d'espansione degli elettroni avviene in direzione verticale (ovvero la direzione data dallo scorrimento delle cariche verso il registro di lettura) rispetto al pixel saturo.

Esiste, infatti, un'elevata differenza di potenziale tra il canale laterale, il quale funge da barriera, e il valore di carica raggiunto con l'accumulo di elettroni.

Tuttavia, è possibile anche avere un blooming nella direzione orizzontale, ma ciò accade quando la condizione di full well si raggiunge nel registro orizzontale ed in seguito a pixel binning.

Visivamente, il blooming di un pixel si presenta come una striscia completamente satura che si estende sopra e sotto il pixel stesso, nella direzione di readout. [1]

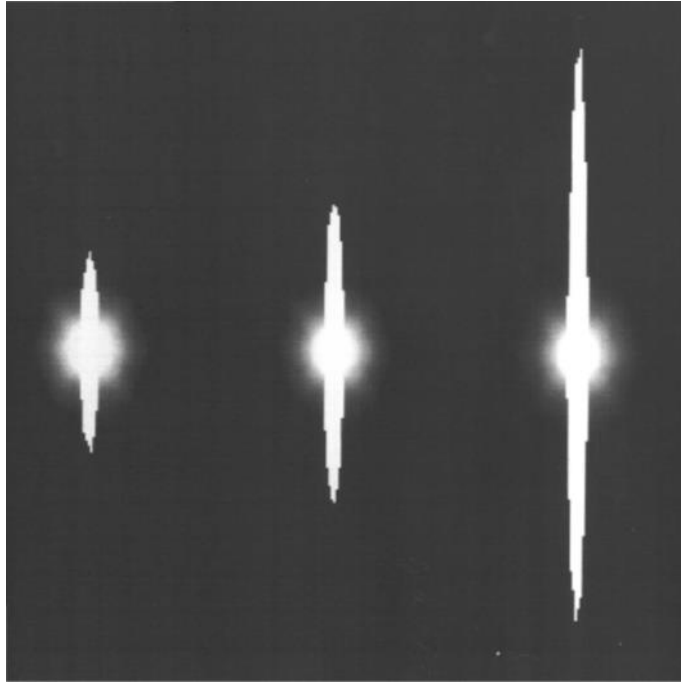


Figura 13 Blooming di tre regioni contenenti pixel saturi. La direzione verticale è quella di readout del sensore CCD [1]

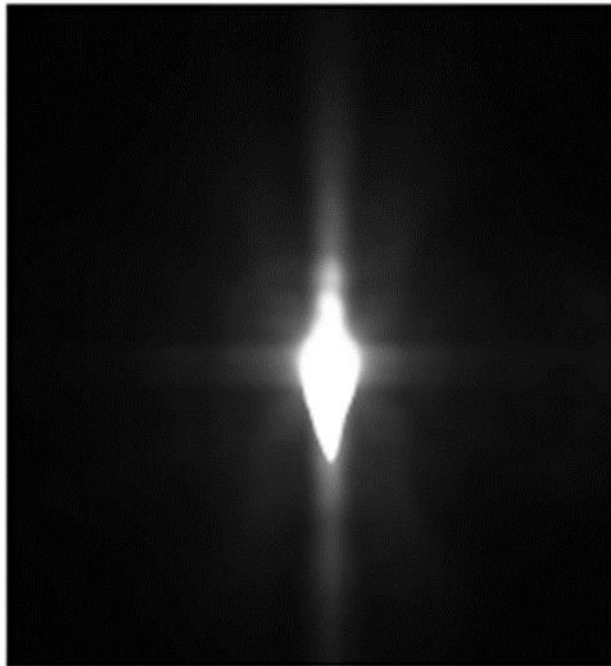


Figura 14 Blooming in un'immagine scattata da MESSENGER [5]

Sono state ideate diverse tecnologie per riuscire ad arginare il blooming, ma quella maggiormente impiegata è l'*Antiblooming Drain*: è possibile infatti prevenire la fuoriuscita incontrollata di elettroni dal pozzo potenziale ponendo dei canali di scarico adiacenti ad ogni singolo pixel oppure, come avviene più comunemente, distribuendoli lateralmente rispetto alle colonne dei pixel, come mostrato in Figura 15.

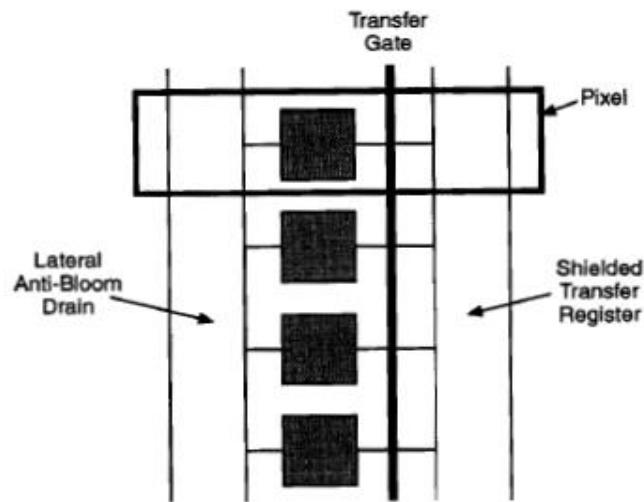


Figura 15 Antiblooming Drain [2]

Così facendo, le cariche in eccesso raccolte nel singolo pozzo potenziale andranno ad accumularsi nel canale di drenaggio prima di fuoriuscire e raggiungere i pixel adiacenti.

Di conseguenza, in fase di lettura non verranno rilevate cariche indesiderate nelle zone prossime ai pixel saturi, permettendo quindi un'acquisizione ed una lettura corretta dei fotoelettroni generati. [2]

### 3.2.1 SIMULAZIONE DELL'EFFETTO DI BLOOMING

Per poter simulare graficamente il rumore di blooming, oltre all'effetto dovuto ai pixel saturi nella direzione di readout, è stato considerato un effetto aggiuntivo (un eccesso di irraggiamento nelle zone adiacenti ai pixel saturi), che segue un andamento di tipo gaussiano attorno a tali pixel.

Pertanto, è stata creata una funzione Matlab che riceve in ingresso la matrice contenente una immagine di partenza  $C$  (priva di blooming), la soglia  $threshold$ , il valore  $sigma$ , ovvero la deviazione standard associata alla distribuzione gaussiana e la costante  $e$  che permette di amplificare l'effetto del rumore nella fase di elaborazione dell'immagine.

Il valore di  $threshold$  rappresenta il limite in Data Number, DN, che può essere anche inteso come il valore finale in scala di grigi in una generica immagine ad 8 bit, oltre il quale il pixel viene considerato saturo.

L'effetto viene applicato prima in una immagine di supporto  $D$  e, per creare tale immagine, è stata introdotta una matrice contenente l'effetto della gaussiana a partire dal suo valore di  $sigma$ : l'effetto di tale distribuzione gaussiana lo si può notare chiaramente nella Figura 16.

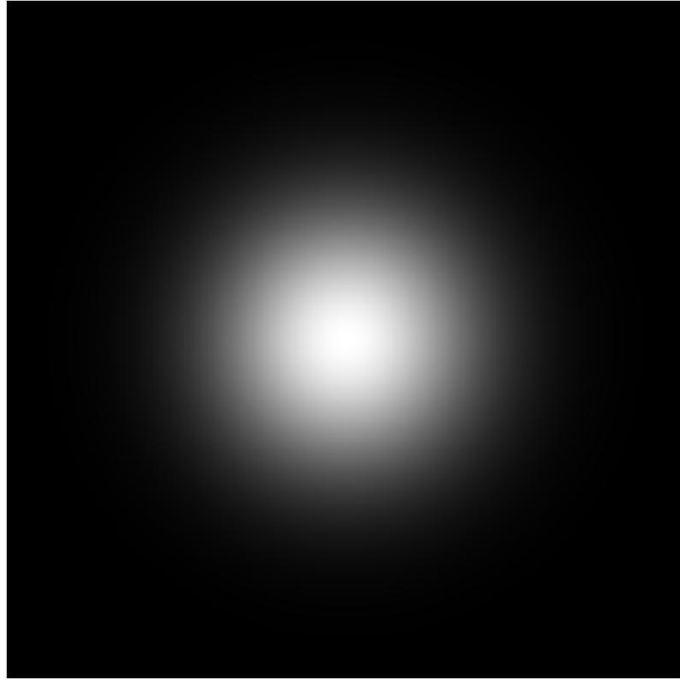


Figura 16 Gaussiana di sigma=50 applicata ad una matrice nera 500x500

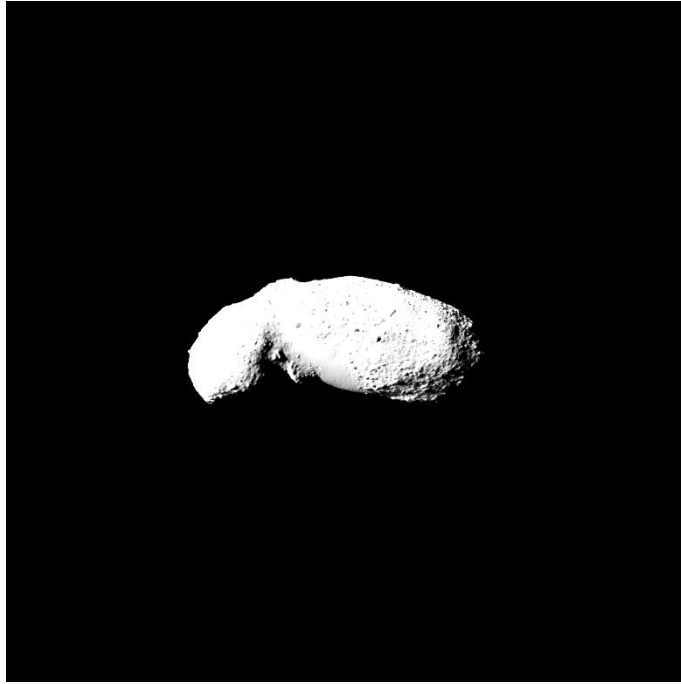
La dimensione di tale matrice è definita da un numero dispari di righe e colonne (size1), per avere il picco della distribuzione perfettamente coincidente con il centro del pixel selezionato.

Per quanto riguarda la funzione di distribuzione gaussiana è stata utilizzata la seguente forma in due variabili:

$$f(x, y) = \left( \frac{1}{2\pi\sigma^2} \right) \exp\left(-\frac{1}{2}\left(\left(\frac{x-\mu_x}{\sigma}\right)^2 + \left(\frac{y-\mu_y}{\sigma}\right)^2\right)\right) \quad (3.7)$$

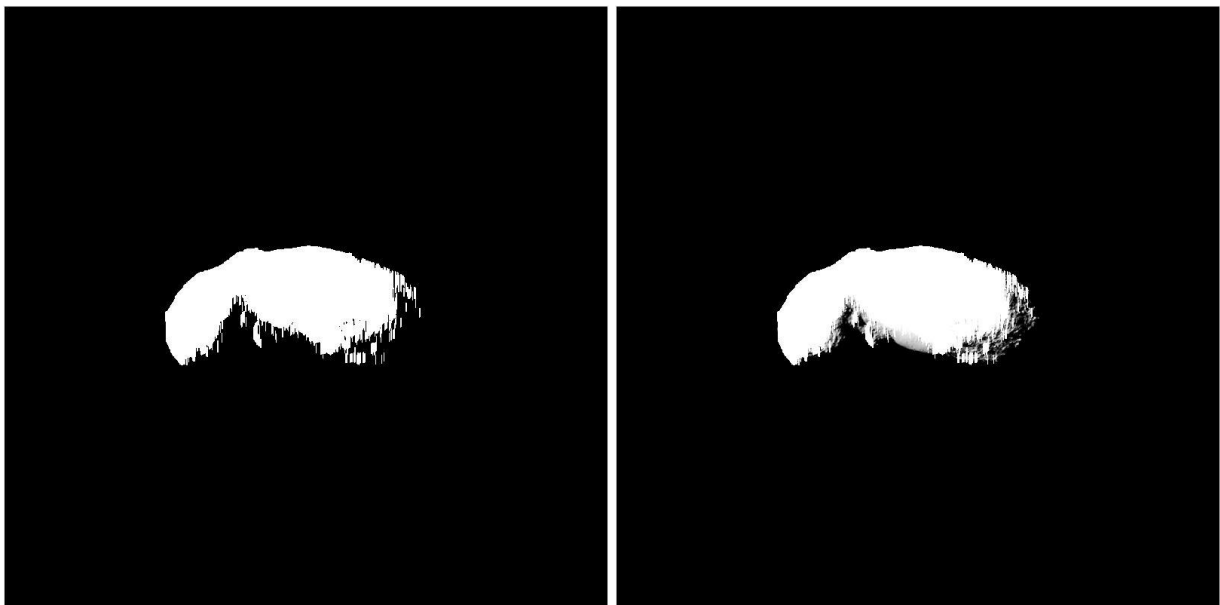
Dove sigma è la deviazione standard desiderata,  $\mu_x$  e  $\mu_y$  rappresentano il centro della gaussiana, il quale viene fatto coincidere con i singoli pixel che superano il valore di soglia della saturazione. Ottenuta quindi la maschera  $D$ , la si applica alla matrice di partenza facendo il massimo tra i valori delle due immagini in ogni pixel, ricavando quindi l'immagine finale raffigurante l'effetto del blooming nell'immagine di partenza.

Poiché l'effetto di blooming ottenuto durante la simulazione era di piccola intensità, si è scelto di presentare come immagine priva di errori la stessa immagine di Itokawa ottenuta con PANGU, ma alla quale è stato aumentato il tempo di esposizione, in modo da avere più pixel saturi e quindi una visualizzazione migliore dell'effetto: la nuova immagine di partenza viene mostrata in Figura 17.



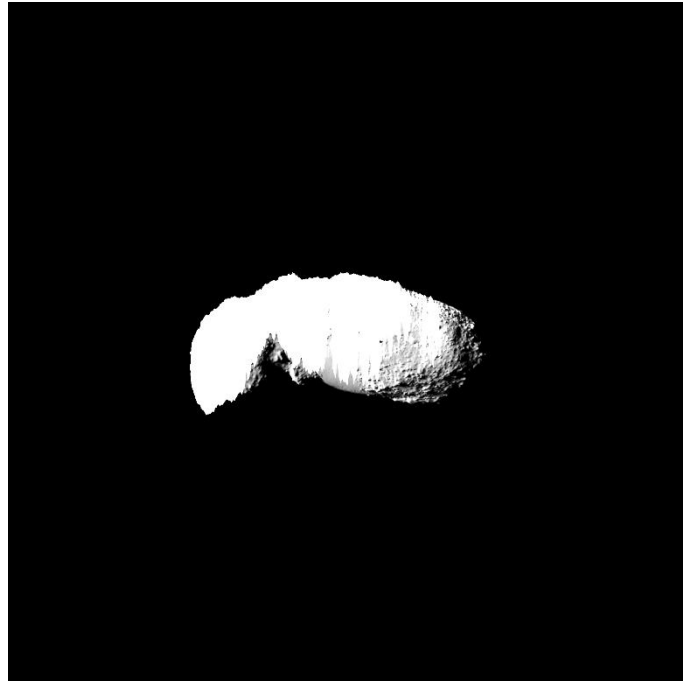
*Figura 17 Immagine di Itokawa con tempo di esposizione maggiore generata con PANGU*

Di seguito vengono mostrate in Figura 18.a la maschera contenente l'effetto di blooming da applicare all'immagine senza rumori esterni, ed in Figura 18.b l'immagine contenente le strisciate verticali dovute al blooming, entrambe ricavate con la simulazione implementata in ambiente MATLAB®.



*Figura 18 a) Maschera contenente solo l'effetto di blooming b) Immagine originale alla quale è stata aggiunta la maschera di blooming (MATLAB)*

Il risultato ottenuto tramite la funzione MATLAB, se confrontato con l'effetto di blooming simulato da PANGU (Figura 19), è molto soddisfacente e questo garantisce correttezza sia per l'analisi teorica, che per l'elaborato pratico di simulazione.



*Figura 19 Effetto di blooming simulato tramite PANGU*

Inoltre, per fornire uno studio completo dell'effetto, vengono riportati i risultati ottenuti implementando in MATLAB anche la componente gaussiana del blooming: nella Figura 20.b si riporta l'immagine contenente sia blooming verticale che l'effetto distribuito, mentre nella Figura 20.a si propone un particolare della maschera applicata all'immagine priva di rumori, al fine di mostrare l'effetto di entrambe le componenti.



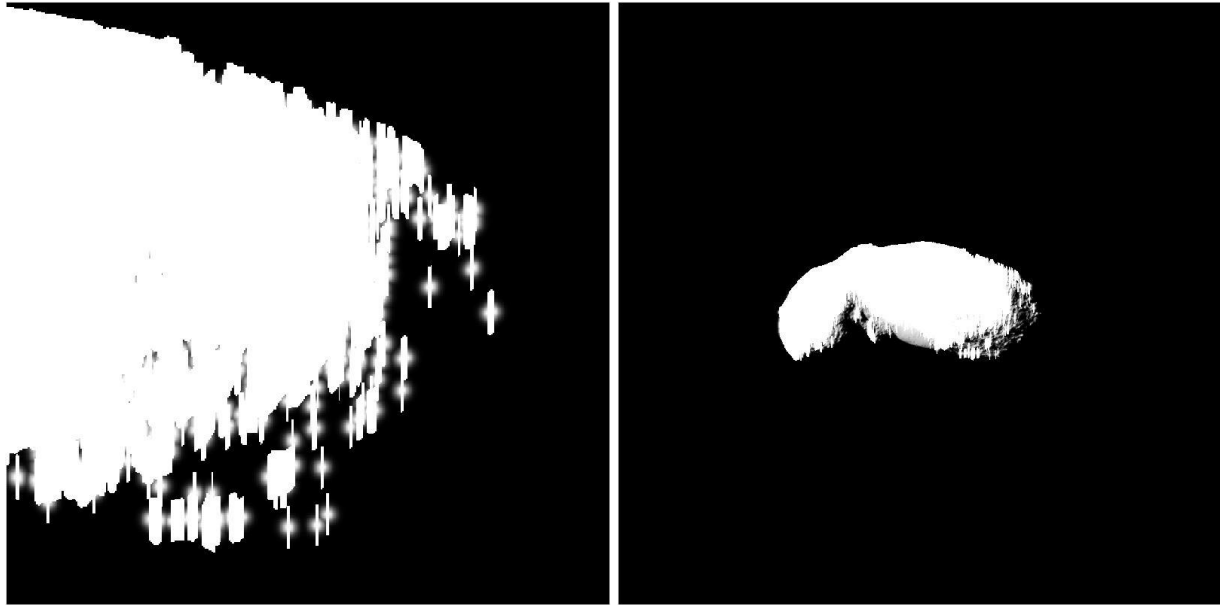


Figura 20 Dettaglio della maschera contenente anche l'effetto di blooming distribuito b) Immagine affetta da blooming verticale e distribuito. Entrambe le immagini sono ottenute dal codice MATLAB

### 3.3 FIXED PATTERN NOISE

La non uniformità nella sensibilità ai fotoni dei singoli pixel è indicata con il nome di *fixed pattern noise*, e viene solitamente espressa come percentuale del numero di fotoelettroni generati dal singolo pixel. [1]

Si può fare una prima distinzione tra le due principali componenti di questo rumore, quella di *Dark Current Fixed Pattern Noise* e la *Photo Response Non-Uniformity (PRNU)*.

Il rumore di Dark Current Fixed Pattern è dovuto al fatto che, durante la fabbricazione dei pixel, vengono introdotte delle imperfezioni nell'area di rilevazione in maniera non omogenea rispetto ai pixel del sensore. [4]

Questo difetto provoca uno stato di eccitamento termico non previsto rispetto ai valori nominali di sensibilità di progetto, dunque il segnale medio di Dark Current non si presenta più uniformemente come analizzato nel capitolo 3.1, ma al contrario presenta valori più elevati distribuiti in maniera randomica, secondo un pattern.

Tuttavia, quando si parla del rumore Fixed Pattern non ci si riferisce in particolar modo a questa componente, in quanto può essere facilmente arginata raffreddando il sensore: provocando differenze di temperatura dell'ordine di una decina di gradi, è possibile dimezzare il valore della densità di dark current prodotta dalla non uniformità. [2]

Il contributo della PRNU è invece più complicato da annullare, in quanto provocato da imperfezioni di costruzione, eventuali presenze di polveri, perdite di micro gocce di olio o da danni successivi all'assemblaggio del sensore.

Il rumore ha infatti luogo durante l'irraggiamento del sensore da parte della sorgente luminosa esterna, dunque l'eccitamento termico provocato non risulta essere solo funzione della temperatura, ma anche del numero di fotoni incidenti.

Questa tipologia di errore non dipende dal tempo, per questo è denominata *Fixed Pattern*: il rumore è invece proporzionale al segnale di input e si traduce si riconosce facilmente in quanto provoca nell'immagine finale una variazione spaziale dell'intensità dei pixel. [4]

Per ridurre l'entità del Fixed Pattern Noise si utilizzano diverse tecniche, ma quella maggiormente considerata è quella chiamata di *Flat Fielding*, la quale sfrutta le caratteristiche lineari dei CCD per creare un algoritmo in grado di uniformare la sensibilità dei pixel allo stesso livello.

$$S_i = \mu_c \frac{b_i}{a_i} \quad (3.8)$$

L'equazione (3.8) è alla base della tecnica di rimozione citata, e mette in relazione il valore medio della calibrazione di Flat Fielding  $\mu_c$ , il valore dell'i-esimo pixel del segnale affetto dal pattern noise  $b_i$  ed il valore, sempre dell'i-esimo pixel, analizzato nella matrice di calibrazione applicata  $a_i$ .

A seguito di questa correzione si riesce ad arginare il problema della non uniformità, ma si registra un significativo aumento di shot noise, prodotto dalla stessa taratura, che provoca un effetto negativo sul rapporto segnale/rumore.

Per avere dunque una stima dell'errore finale da considerare bisogna introdurre le componenti di shot noise relative ai pixel contenenti rumore e a quelli di calibrazione (rispettivamente  $\sigma_{b_i}$  e  $\sigma_{a_i}$ ), ed il rumore di lettura  $\sigma_R$  [1]:

$$\sigma_{S_i}^2 = \sigma_{b_i}^2 + \left(\frac{b_i}{a_i} \sigma_{a_i}\right)^2 + \sigma_R^2 \quad (3.9)$$

Questo processo va implementato appena si ricava l'immagine affetta da rumore dal sensore CCD in quanto, non agendo immediatamente, si ricommetterebbe un errore sull'immagine causato da un residuo di Fixed Pattern Noise.

Inoltre, è possibile quantificare il dato relativo alla non uniformità di ogni pixel ( $P_{Ni}$ ), a partire dal segnale generato e dal valore della varianza del rumore indicato in equazione (3.10):

$$P_{Ni} = \frac{\sigma_{Si}}{S_i} \quad (3.10)$$

### 3.3.1 SIMULAZIONE DEL FIXED PATTERN NOISE ED ANALISI DEI RISULTATI

Anche in questo caso si è utilizzato il programma MATLAB® per avanzare una simulazione del Fixed Pattern Noise.

Come primo passaggio, si è generata una matrice di valori che rappresentano la non uniformità di sensibilità relativa ad ogni singolo pixel. Tale matrice sta alla base dei valori che devono essere assunti dalla calibrazione di Flat Fielding, atta a diminuire l'effetto dell'errore introdotto.

Pertanto, la funzione richiede come primo dato di input la matrice rappresentante l'immagine non affetta da rumore, e nel nostro caso si è inserita la matrice dell'immagine di Itokawa, mentre l'altro valore richiesto all'utente è il rapporto tra numero di fotoelettroni ed il corrispondente numero di Data Numbers (nella simulazione si è utilizzato il valore della missione MESSENGER che prevedeva un DN ogni 15 fotoelettroni).

Il dato sulla non uniformità è stato ottenuto attraverso una distribuzione gaussiana di valori percentuali tipici (inferiori al 10%), in modo da simulare una sensibilità dei pixel appartenente all'intervallo [90%, 110%].

Si è quindi creata una matrice  $r$  con valori ottenuti dalla distribuzione gaussiana e randomica.

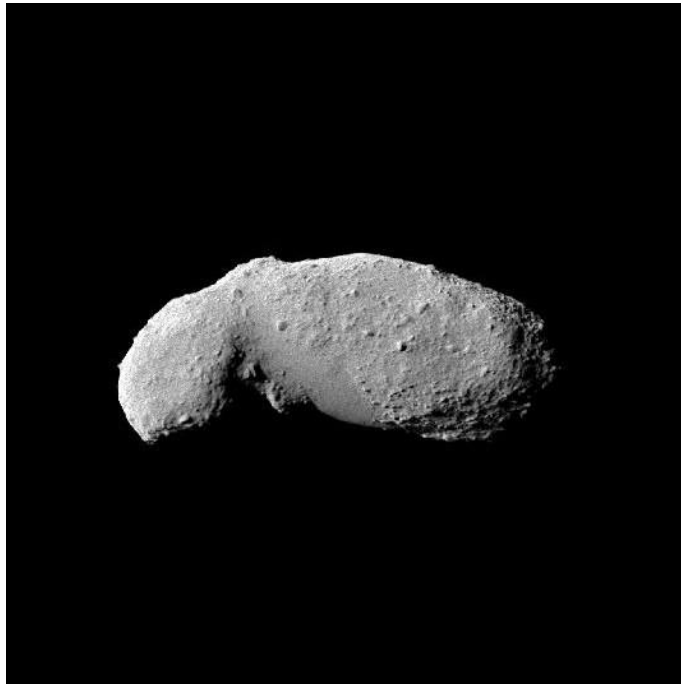
Poiché per l'analisi matematica sono richieste informazioni riguardo il numero di elettroni, si è moltiplicato il valore dei DN per il guadagno di conversione richiesto in input: a questo punto si è applicata la matrice  $r$  e si è ottenuta la matrice  $NU$  contenente il numero di fotoelettroni generati con il rumore di Fixed Pattern Noise.

Affinché si potesse visualizzare l'immagine affetta da rumore, si è scalato nuovamente il numero di elettroni in DN e si è ottenuta la matrice  $raw$ , primo output di funzione.

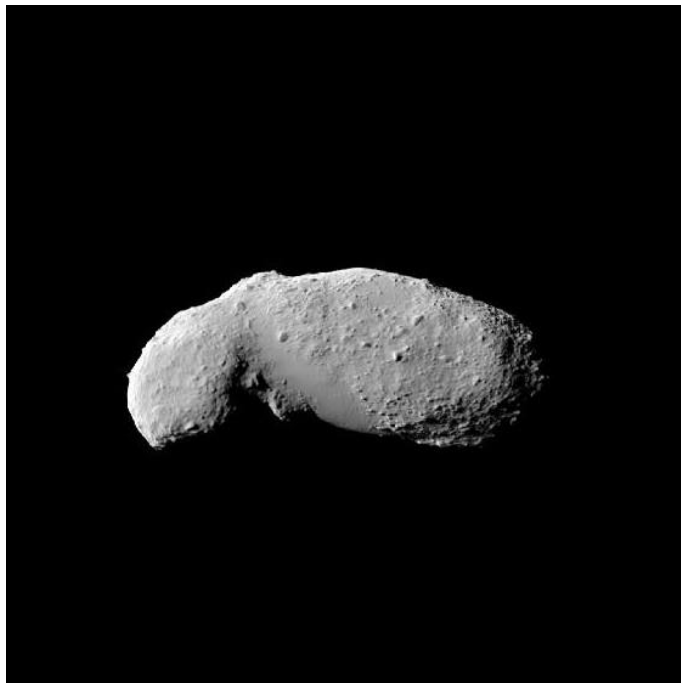
Avendo ora a disposizione l'immagine che sarebbe ricavata dalla lettura del CCD, si è calcolato il contributo della calibrazione d'immagine necessaria, in modo da avere a disposizione tutti i dati richiesti dall'equazione (3.8), implementata considerando  $\mu_C$  pari al valore medio della matrice di Flat Fielding appena ricavata (chiamata *flat*).

Si procede ricavando il segnale pulito dall'errore di Fixed Pattern Noise attraverso l'equazione (3.8) e, utilizzando tre valori realistici associati alle tre componenti di shot noise e read noise, si è specificato il dato sulla non uniformità dei pixel con l'equazione (3.10).

Vengono di seguito riportate: in Figura 21 l'immagine contenente il rumore ed in Figura 22 l'immagine prodotta dopo l'applicazione della calibrazione Flat Fielding.



*Figura 21 Immagine affetta da Fixed Pattern Noise generata con MATLAB*

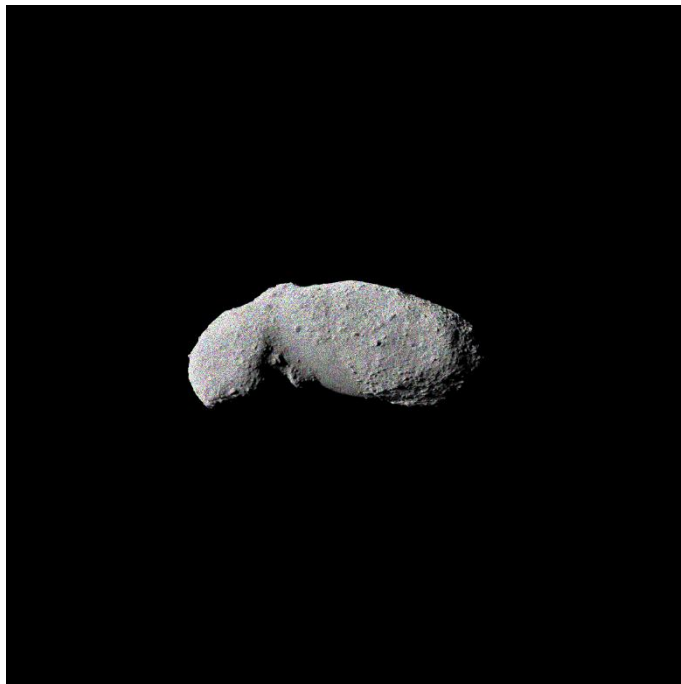


*Figura 22 Immagine ottenuta dopo la calibrazione Flat Fielding (MATLAB)*

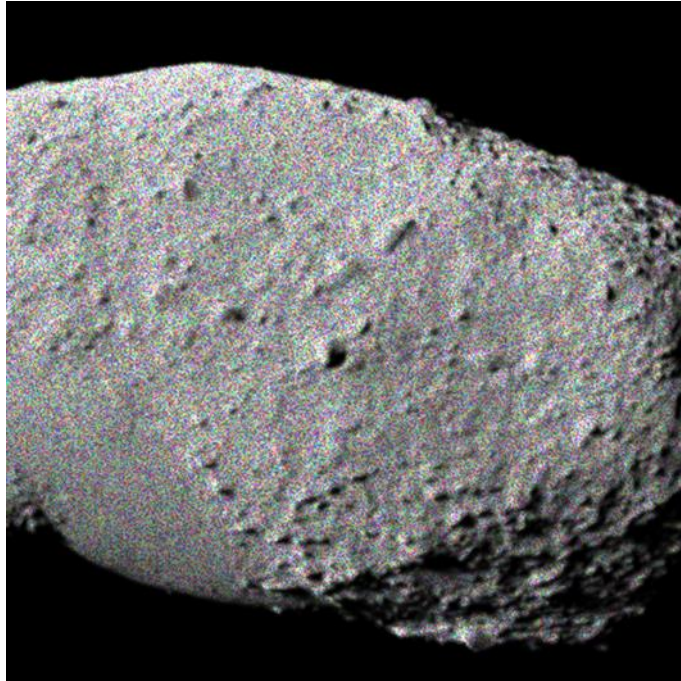
L'analisi di questo rumore è stata effettuata anche con il software PANGU attraverso l'apposita funzione per la simulazione dell'errore di PRNU, ma non per ottenere informazioni sul grado di non uniformità del pixel, quanto più per avere un riscontro sulla generazione dell'immagine affetta dal Fixed Pattern Noise.

L'unica differenza riscontrabile visivamente è rappresentata dal fatto che nel caso dell'immagine generata dal codice MATLAB, il rumore è monocromatico, andando a rispecchiare la natura del sensore CCD privo di filtri, mentre nel caso dell'immagine ottenuta con PANGU il rumore introdotto è nei canali rosso, verde e blu in maniera non uniforme, discostandosi quindi dal caso di sensore monocromatico senza filtri.

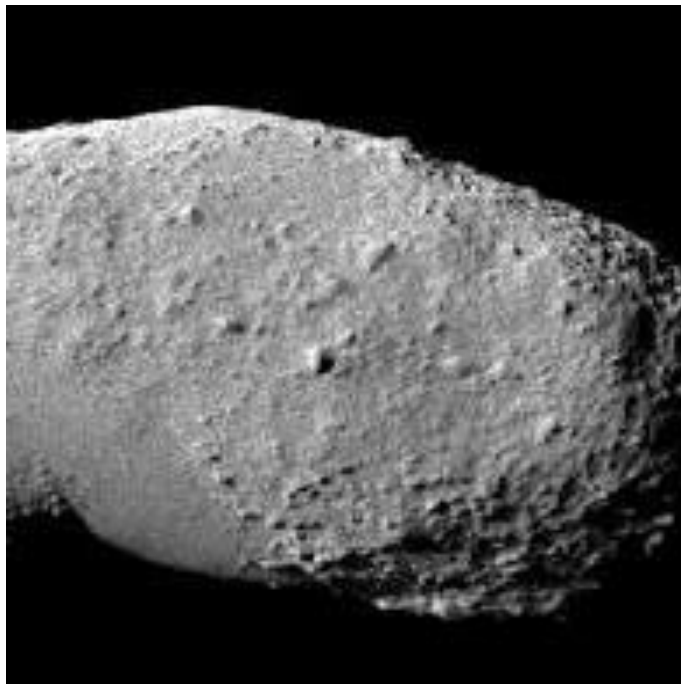
Viene inoltre fornito in Figura 24 un particolare dell'immagine ottenuta tramite PANGU (Figura 23), per mostrarne meglio l'effetto del rumore espresso in canali rosso, verde e blu e di come questo si differenzi da quello monocromatico, che rispecchia maggiormente il caso specifico di un sensore senza filtri colorati ottenuto con il codice MATLAB®.



*Figura 23 Immagine affetta da Fixed Pattern Noise simulata con PANGU*



*Figura 24 Dettaglio della Figura 23*



*Figura 25 Dettaglio dell'immagine con Fixed Pattern Noise ottenuta tramite MATLAB*

### 3.4 READOUT SMEAR NOISE

Il Readout Smear Noise è un tipo di rumore che si verifica nelle architetture dei CCD durante la lettura delle cariche elaborate dal sensore, quindi durante lo scorrimento di queste dall'area di generazione dei fotoelettroni fino al registro di lettura.

Poiché la lettura delle cariche avviene colonna per colonna (o riga per riga a seconda dell'orientamento del sensore), durante il loro scorrimento il sensore è ancora in grado di raccogliere fotoelettroni.

In mancanza di elementi oscuratori come gli otturatori meccanici (*shutter*), il sensore risulta essere esposto per maggior tempo all'incidenza dei fotoni: tutto ciò si traduce in un aumento dei valori d'intensità dei pixel, fortemente dipendente dalla posizione della colonna considerata. [6]

La problematica appena descritta (ovvero l'effetto dovuto alla mancanza di otturatore meccanico) prende il nome di Electronic Shutter Effect.

Risulta evidente che l'intensità di questo errore indotto dall'electronic shutter è proporzionale al rapporto tra il tempo di esposizione del sensore alla sorgente luminosa e il tempo di trasferimento delle cariche: se questi sono dello stesso ordine di grandezza comparirà un effetto di *smear*, ovvero di questo effetto di "strisciata" dell'immagine originale.

Lo smear può essere descritto nel seguente modo: considerato un generico sensore, un pixel con coordinate  $(x, y)$  raccoglie, a meno di altri tipi di rumore, un certo numero di fotoelettroni proporzionale al tempo di esposizione del pixel stesso.

Mentre l'immagine scorre di una riga verso il registro di lettura, moto che nel nostro esempio consideriamo diretto nel verso negativo delle  $y$ , la carica accumulata dal pixel  $(x, y)$  si sposterà in posizione  $(x, y - 1)$ .

In posizione  $(x, y - 1)$ , la carica totale sarà pari a quella dei fotoelettroni generati dal tempo di esposizione dell'immagine originale, a cui sono aggiunti i fotoelettroni della posizione  $(x, y - 1)$  generati da un tempo di esposizione proporzionale al tempo di scorrimento di una riga verso il registro di lettura.

La raccolta indesiderata di elettroni si verifica fintanto che la riga in questione non raggiunge il registro di lettura: l'immagine realmente acquisita può essere dunque interpretata come la somma dell'immagine desiderata e della stessa immagine progressivamente traslata nel verso dato dalla direzione di lettura.

Solitamente questo processo avviene in tempi dell'ordine dei microsecondi, ad esempio, per la missione MESSENGER, il tempo di trasferimento di una singola colonna era equivalente a 0.976 ms. [3]

Sostituendo invece un'architettura Full Frame con un Frame Transfer Imager, si riesce ad arginare la generazione dello smear noise: questo modello presenta due arrays identici di cui il

primo destinato alla raccolta di fotoni e il secondo ricoperto da uno shutter metallico atto a proteggere la fase di accumulazione delle cariche (Figura 26).

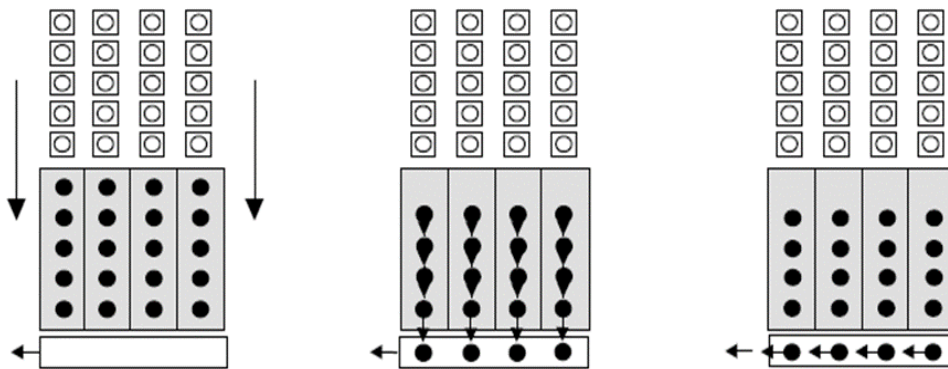


Figura 26 Funzionamento del Frame Transfer Imager [2]

Ciò richiede un tempo di trasferimento di circa  $500 \mu\text{s}$ , mentre il passaggio successivo da questa zona di raccolta fino al registro di lettura richiede solitamente  $64 \mu\text{s}$  per ogni riga. [14]

Un'ulteriore differenza tra gli errori generati nei due modelli di CCD riguarda le caratteristiche dello smear, che si verifica a causa della diversa strategia di lettura.

Con il Full Frame si ottiene la strisciata descritta all'inizio del capitolo e dalla Figura 27, ovvero, ipotizzando la lettura verso il basso, uno smear che si articola dal pixel con l'intensità in questione fino all'ultima riga in alto.

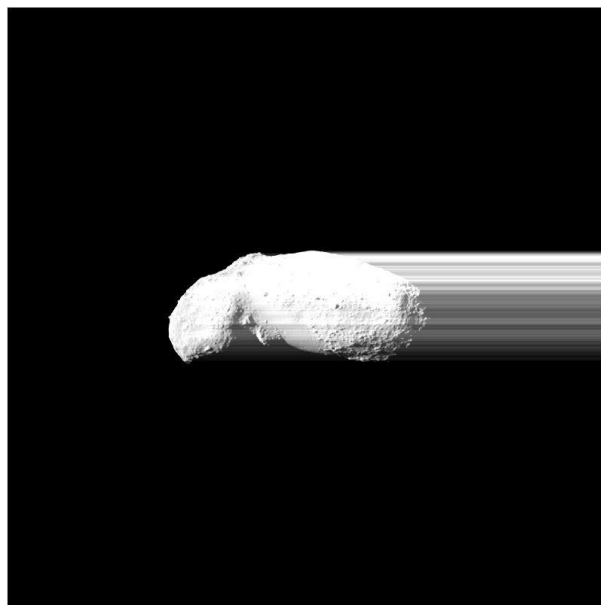


Figura 27 Readout Smear Noise, immagine ottenuta durante la simulazione MATLAB tramite applicazione del rumore ad un'immagine pulita



Utilizzando invece il Frame Transfer Imager si verifica un primo spostamento nell'area di accumulazione e poi una lettura classica riga per riga, provocando un effetto di smear sia al di sotto che al di sopra (in direzione della lettura) del pixel luminoso.

### 3.4.1 SIMULAZIONE DEL RUMORE DI READOUT SMEAR ED ANALISI DEI RISULTATI OTTENUTI

Di seguito viene riportato il procedimento seguito durante la creazione della funzione in MATLAB® rappresentante questa tipologia di rumore, il cui output è mostrato in Figura 27.

Come dati di input vengono inseriti il tempo di esposizione previsto dal modello di camera scelto  $t_{exp}$ , quello di trasferimento di una singola colonna durante il procedimento di lettura  $t_{col}$  e la matrice dell'immagine pulita  $A$ : nel nostro esempio sono stati considerati i dati relativi a un sensore dello strumento Mercury Dual Imaging System (MDIS), della missione MESSENGER. [3]

$$\begin{cases} t_{exp} = 9.8 \text{ s} \\ t_{col} = \frac{1}{1024} \text{ s} \end{cases}$$

Successivamente vengono create tre matrici con le stesse dimensioni di quella in ingresso di cui la prima ( $A_1$ ) contiene i valori d'intensità dei pixel della matrice in ingresso normalizzati per il tempo di esposizione, ovvero valore d'intensità luminosa che riceve ogni pixel durante l'unità di tempo di esposizione.

La seconda ( $A_2$ ) è ottenuta moltiplicando ogni riga della matrice appena descritta per il tempo di trasferimento di colonna e da questa si prende l' $i$ -esima colonna e si aggiunge nell'ultima matrice ( $B$ ) a partire dalla posizione successiva a quella considerata: la matrice  $B$  rappresenta la maschera contenente l'effetto dello smear da applicare ai valori dell'immagine iniziale.

Questo procedimento permette di ricreare fedelmente l'effetto dello smear noise in quanto ogni lettura di una singola colonna lascia un contributo d'intensità non nulla che si ripete da quella colonna in poi per tutta l'immagine.

Con i dati presentati in precedenza, l'effetto del rumore di smear non era apprezzabile e la correzione calcolata aveva come valore massimo zero (Figura 28).

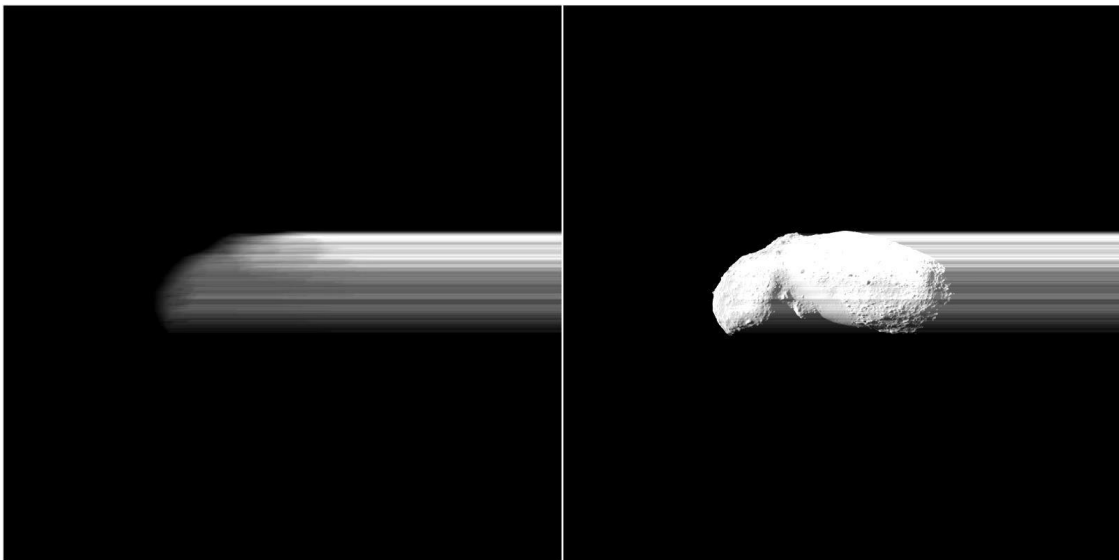
Per poter avere un effetto apprezzabile, è stato artificialmente scalato il tempo di trasferimento di colonna di un coefficiente correttivo  $k$ : l'effetto di tale coefficiente può essere

interpretato come un aumento del tempo che impiega il sensore a processare una singola riga, aumentando, pertanto il valore di esposizione corrispondente di ogni singola riga in fase di readout.

Dapprima è stato associato a  $k$  il valore 27 (e con questo si è ottenuta l'immagine finale in Figura 27 riportata anche in Figura 29) e poi un valore pari a 40 (Figura 30), in modo da poter notare la differenza che si presenta con tempi di trasferimento crescenti.



*Figura 28 Maschera d'errore ed immagine finale nel caso reale ( $k=1$ )*



*Figura 29 Maschera d'errore ed immagine finale ( $k=27$ )*

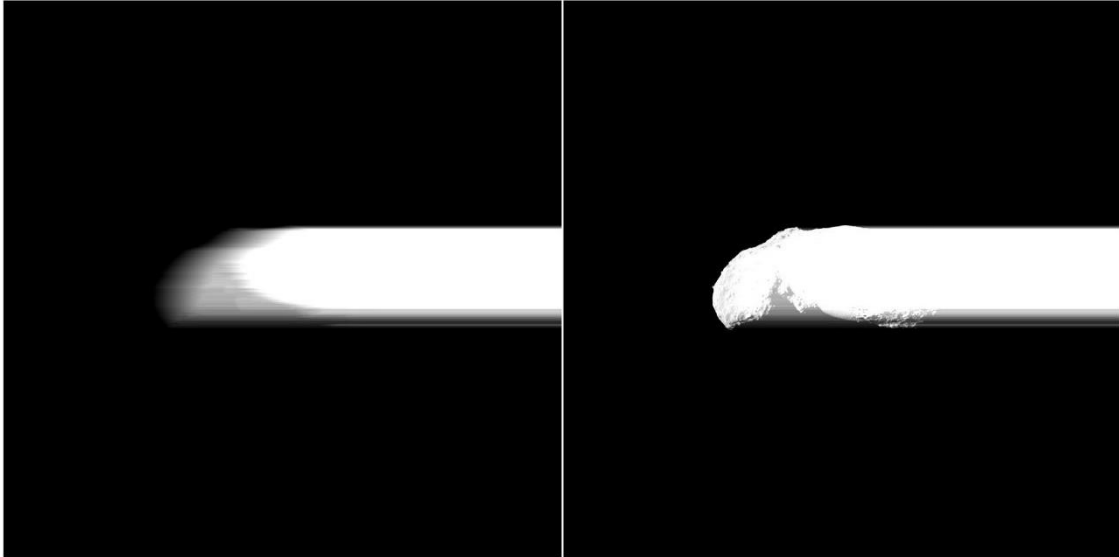


Figura 30 Maschera d'errore ed immagine finale ( $k=40$ )

Nel caso della Figura 29, l'effetto dello smear è meno intenso del caso riportato in Figura 30, poiché il parametro di scala del tempo associato allo spostamento di una colonna è in rapporto pari a 27/40 dell'ultimo caso: è immediato notare l'effetto tale per cui la parte sinistra della maschera risulti meno intensa della parte opposta a causa della modalità di sviluppo dello smear noise.

I risultati ottenuti sono perfettamente in linea con la simulazione operata attraverso PANGU che, grazie alla specifica funzione *smear*, permette di visualizzare l'effetto di questo rumore di lettura come indicato in Figura 31.

La similarità tra l'output di PANGU e quello di MATLAB®, permette di considerare la funzione creata una buona approssimazione dell'effetto reale del rumore.

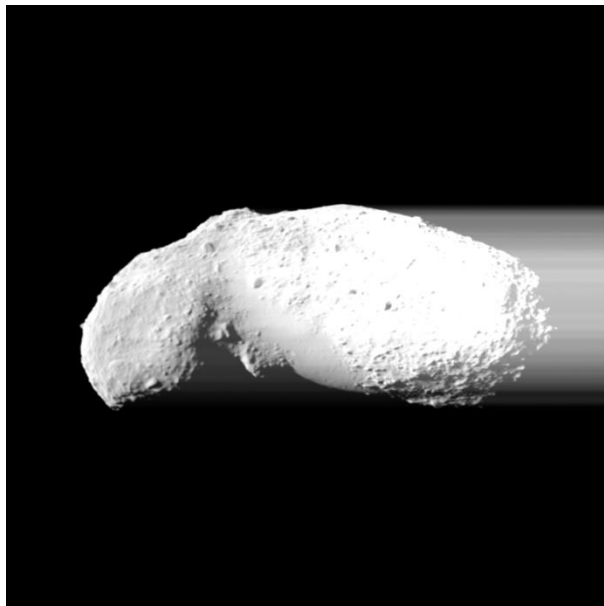


Figura 31 Effetto di Readout Smear Noise simulato con PANGU

### 3.5 ADC QUANTIZATION NOISE

Benché i sensori fotoelettrici restituiscano in output un'immagine digitale, il punto di partenza del loro processo di lavorazione è un dato riguardante il numero di fotoelettroni generati a partire da un flusso di fotoni incidente: questa non è un'informazione ottenuta digitalmente, ma si rispecchia in un segnale analogico difficilmente interpretabile come immagine.

La conversione necessaria a trasformare il dato di tipo analogico in dato digitale viene chiamata quantizzazione, operazione possibile tramite un componente elettronico, l'*Analog-to-Digital Converter* (ADC chip).

Esso è adiacente al registro di lettura per quanto riguarda i sensori CCD, mentre è collocato all'interno di ogni pixel nella configurazione CMOS.

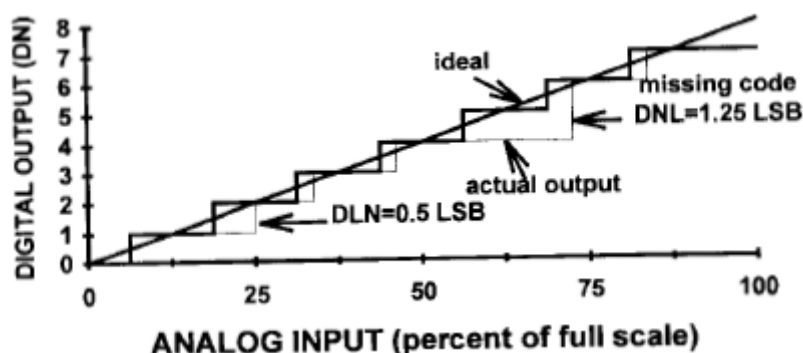


Figura 32 Output digitale di un segnale analogico [2]

Il processo di quantizzazione prevede di dividere una grandezza di partenza a valori continui in un numero finito di valori discreti; il numero di valori rappresentabili dalla quantizzazione dipende dal valore di quantizzazione scelto. Quest'ultimo rappresenta il numero di bit con il quale si intende digitalizzare il segnale, e definisce il massimo numero di intervalli discreti a cui sono associati i valori analogici di partenza.

Il numero di intervalli, quindi, dipende dal numero di bit  $N_{BITS}$  ed è qui indicato con  $N_{ADC}$  (espresso in Data Number, DN):

$$N_{ADC} = 2^{N_{BITS}} \quad (3.11)$$

A questo punto si può individuare il valore che assume il Last Significant Bit (LSB), ovvero il bit meno significativo, in termine della grandezza che viene quantizzata nella conversione. Tale

valore può essere ottenuto dividendo la capacità massima del pozzo potenziale, espressa in numero di elettroni ( $N_{well}$ ), per il numero totale di livelli di quantizzazione:

$$LSB = \frac{N_{well}}{N_{ADC}} \quad (3.12)$$

Il termine di guadagno della conversione permette di passare dal valore analogico in ingresso al valore quantizzato, ed è spesso indicato a partire dai valori di voltaggio piuttosto che dall'espressione dei numeri di elettroni corrispondenti. Perciò, il guadagno può essere espresso come il numero di livelli di quantizzazione totali divisi per la differenza tra voltaggio massimo e minimo riscontrabili, rispettivamente  $V_{max}$  e  $V_{min}$ . [4]

$$G = \frac{N_{ADC}}{V_{max} - V_{min}} \quad (3.13)$$

Ogni volta che il valore assunto da un pixel viene quantizzato, si introduce intrinsecamente un'incertezza nel valore di partenza e questo rappresenta l'*ADC Quantization Noise*. Il rumore prodotto è presente in ogni pixel, dove il valore di fotoelettroni indicato non corrisponde al vero numero di fotoelettroni effettivamente generati nel segnale analogico: il valore discretizzato di fotoelettroni  $N_{pe}$  è quindi funzione del numero di elettroni iniziale e dell' *LSB*.

$$N_{pe(ADC)} = LSB * floor \left[ \frac{N_{pe}}{LSB} \right] \quad (3.14)$$

L'errore introdotto dall'ADC è funzione della capacità del pozzo potenziale, ed è possibile esprimerlo sia in funzione della sua tensione  $V_{NOISE}$ , sia del numero di elettroni  $N_{NOISE}$ . [2]

$$V_{NOISE} = \frac{V_{LSB}}{\sqrt{12}} \quad (3.15)$$

$$N_{NOISE} = \frac{N_{well}}{N_{ADC} \sqrt{12}} \quad (3.16)$$

Nell'equazione (3.15),  $V_{LSB}$  rappresenta il valore di tensione assegnato al singolo *LSB*, quindi alla distanza che intercorre fra i singoli valori quantizzati.

### 3.5.1 ANALISI SULL'INTENSITA' DEL RUMORE DI ADC

Data la natura stessa del calcolatore, è impossibile considerare un valore da esso generato come analogico: per questo motivo un qualsiasi input utilizzato per rappresentare questo tipo di rumore sarebbe già di natura digitale e verrebbe quindi a mancare il principio fisico che sta alla base di questa conversione.

L'approccio che si intende utilizzare per l'analisi dell'ADC Quantization Noise è diverso da quello adottato in precedenza in quanto l'obiettivo è quello di rappresentare l'intensità del rumore per poterne capire l'influenza in un contesto d'applicazione realistico, e non di ottenere una sua simulazione grafica.

Pertanto, per simulare un segnale analogico, verrà utilizzato un segnale digitale, ma caratterizzato da un numero molto maggiore di valori di quantizzazione rispetto a quello impiegato nell'uscita dalla conversione ADC, tale per cui esso possa essere considerato al pari di un segnale continuo. Per farlo si passa all'analisi della funzione creata per la simulazione del rumore, sempre attraverso il programma MATLAB®.

I primi due valori da inserire in ingresso nella funzione creata per la simulazione sono  $N_{well}$  e  $N_{bit}$  che, come indicato nelle formule impiegate per la descrizione del rumore, rappresentano rispettivamente il numero di fotoelettroni di capienza massima dei pozzi potenziali e il numero di bit utilizzati per la quantizzazione.

Inoltre, vengono richiesti il  $gradient_{value}$ , ovvero la percentuale con la quale si vuol far crescere l'intensità del gradiente utilizzato per il segnale iniziale, ed il dato  $N_{pixel}$  che se specifica la dimensione dell'immagine che vogliamo ottenere.

Per rendere la simulazione di questo rumore realistica, sono stati scelti come valori d'input quelli previsti dalla conversione ADC e dal sensore CCD della missione spaziale americana MESSENGER [14]:

$$\begin{cases} N_{well} = 360000 \\ N_{bit} = 8 \end{cases} \quad (3.17)$$

Ai fini della quantizzazione però è necessario conoscere il valore di  $N_{well}$  non in termini di fotoelettroni, ma nel numero di Data Numbers associati: per la missione MESSENGER questa conversione prevedeva 1 DN ogni 15 fotoelettroni, dunque il dato che verrà impiegato sarà  $DN_{well} = 24000$ .

Essendo in ogni caso un valore così elevato, è legittimo considerare questo segnale discreto come un segnale continuo ai fini dell'analisi: al momento possediamo quindi una matrice  $A_{analog}$  di Data Number con valori possibili fino a 24000, per ora vuota.

Procediamo applicando a questa matrice un gradiente, facendo sì che alla colonna più a sinistra sia associato il valore di  $DN = 0$  e che l'intensità cresca secondo quanto indicato dal valore  $gradient_{value}$  inserito in ingresso.

Tuttavia, non è possibile mostrare l'intero range dinamico di tale matrice a causa del limite di rappresentazione dell'immagine che impone la visualizzazione di matrici con valori compresi tra 0 e 255.

Giunti a questo punto si crea una matrice quantizzata  $A_{quant}$  tramite una conversione lineare, associando ad ognuno dei valori della matrice precedente, un altro intero compreso nell'intervallo di valori da 0 a 255 (quindi considerando 8 bit di quantizzazione), ottenuto tramite troncamento:

$$A_{quant} = floor \left( A_{analog} * \frac{255}{DN_{well}} \right) \quad (3.18)$$

Il risultato ottenuto dopo la quantizzazione ha dunque un range dinamico che è tipico delle immagini digitali, pertanto esprimibile su monitor (Figura 33)



Figura 33 Matrice  $A_{quant}$ , con gradiente che si sviluppa in direzione orizzontale, quantizzato a 8 bit

La quantizzazione applicata introduce un errore (dovuto al troncamento) rispetto al segnale originale ( $A_{analog}$ ), e per mostrarlo si è scelto di rappresentare l'andamento dei valori DN per una singola riga, dato che il gradiente si estende nella direzione delle righe.

Quindi, è opportuno riottenere il valore in DN del segnale quantizzato, andando ad applicare l'inverso del fattore di scala utilizzato per ottenere la matrice  $A_{quant}$ :

$$A_{quantDN} = A_{quant} * \frac{DN_{well}}{255} \quad (3.19)$$

In Figura 34 è rappresentato l'andamento di una generica riga sia per la matrice  $A_{analog}$  (Figura 34.a) che per la  $A_{quant}$  (Figura 34.b), al fine di visualizzare l'effetto di quantizzazione del segnale.

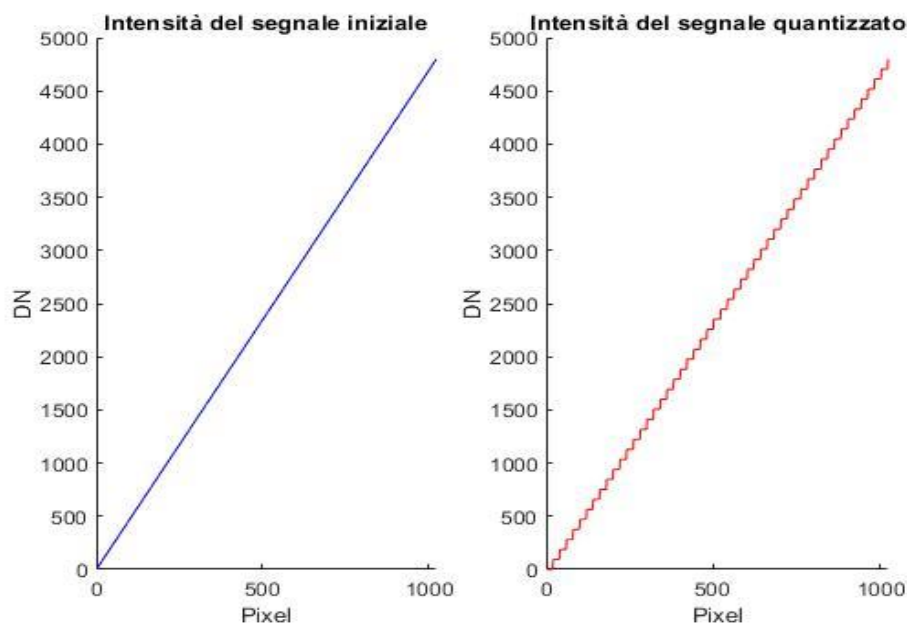


Figura 34 a) DN segnale iniziale ( $A_{analog}$ ) b) Segnale quantizzato a 8 bit riconvertito in DN ( $A_{quantDN}$ , valori della sonda MESSENGER)

L'errore di quantizzazione che viene commesso dall'ADC converter viene quindi rappresentato dalla differenza fra la matrice del segnale continuo di partenza ( $A_{analog}$ ) con il valore quantizzato e nuovamente scalato,  $A_{quantDN}$ .

$$Error = A_{analog} - A_{quantDN} \quad (3.20)$$



Poiché l'errore espresso in DN non è rappresentabile come immagine, si mostra il suo andamento nel caso di una riga generica, come mostra la Figura 35.

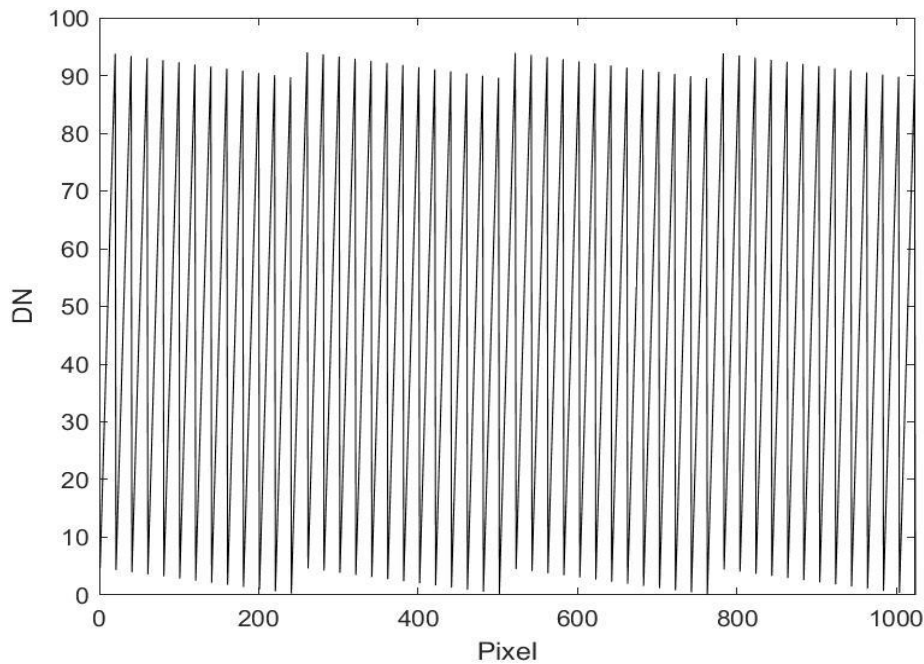


Figura 35 Andamento dell'errore di quantizzazione (*Error*) espresso in DN per i valori di una riga (valori tipici della sonda MESSENGER)

Poiché l'immagine di partenza è un gradiente lineare, si ottiene l'andamento dell'errore di quantizzazione di Figura 35, dove il valore dei suoi massimi coincide con il LSB, che nel nostro caso è pari a 94: la funzione di arrotondamento utilizzata è una funzione di troncamento, dunque l'errore commesso risulta sempre positivo o al più nullo.

Tuttavia, per ottenere un effetto apprezzabile dal range dinamico con cui le immagini possono essere mostrate (quindi un effetto apprezzabile all'interno di una immagine ad 8 bit), si è scelto di ripetere la stessa analisi, andando a cambiare il valore di  $N_{well}$ ,  $N_{bit}$  di quantizzazione e di  $gradient_{value}$ .

I valori scelti in ingresso alla funzione sono  $N_{well} = 256 * 15$  (dove 15 è il rapporto fra fotoelettroni accumulati per Data Number) utilizzando una quantizzazione a 4 bit e  $gradient_{value} = 1$ , in tal modo è possibile ottenere una matrice  $A_{analog}$  rappresentabile su monitor (a valori nel range [0 255] ), e una matrice  $A_{quantDN}$  in cui è visibile l'effetto a bande introdotto dalla quantizzazione a 4 bit (Figura 36.b).

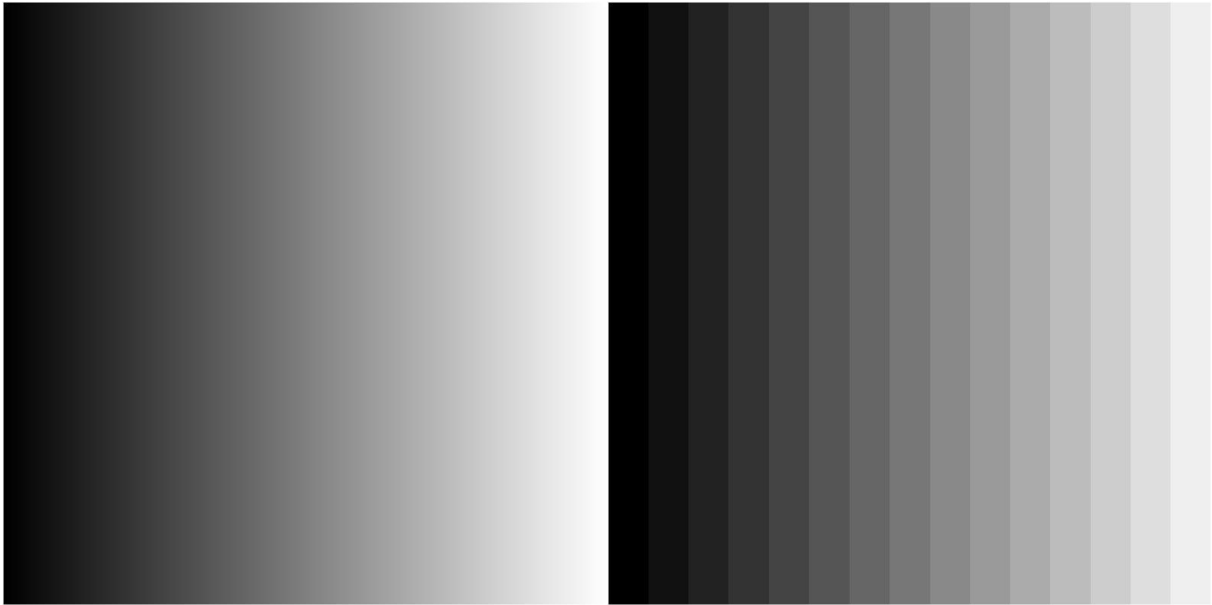


Figura 36 a) Segnale ad 8 bit ( $A_{analog}$ ) b) Segnale quantizzato a 4 bit ( $A_{quantDN}$ )

In questo caso è possibile riottenere l'andamento a gradini come in Figura 34, considerando la Figura 37.a come una riga dell'immagine reale a valori continui ( $A_{analog}$ ) e la Figura 37.b come la stessa riga della matrice quantizzata ( $A_{quantDN}$ ). **Errore. L'origine riferimento non è stata trovata.**

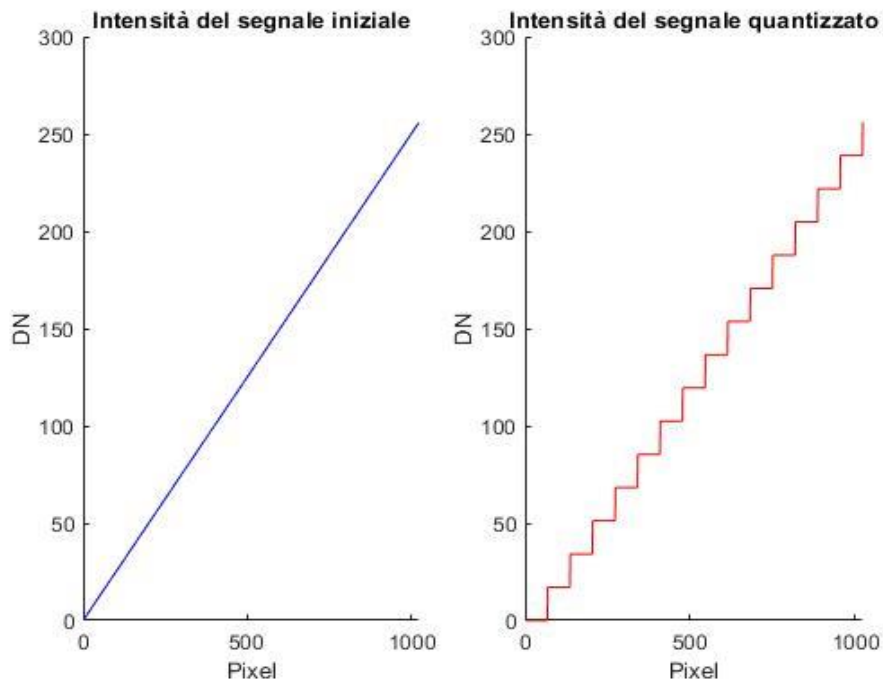


Figura 37 a) Segnale iniziale di una riga, espresso in DN ( $A_{analog}$ ) b) Segnale quantizzato a 4 bit espresso in DN ( $A_{quantDN}$ )

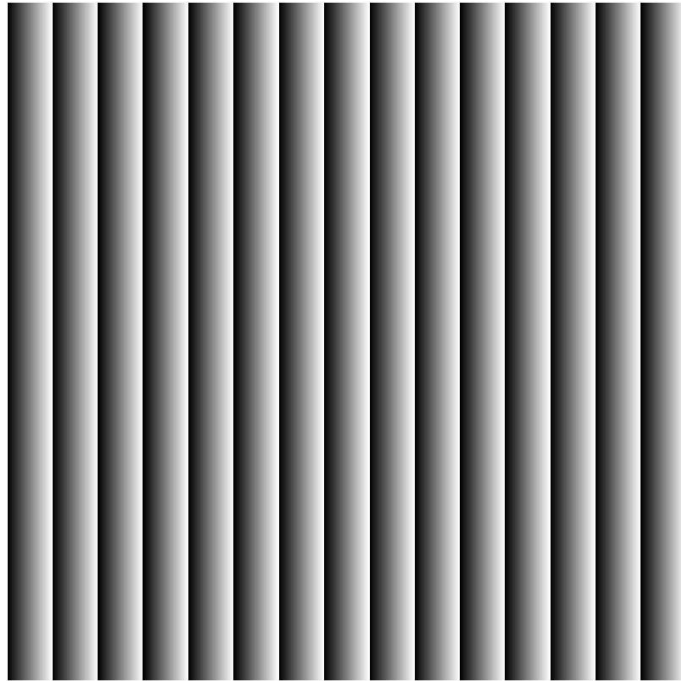


Figura 38 Errore di quantizzazione nel caso a 4 bit con valore massimo normalizzato al valore di bianco (255)

In Figura 38 viene mostrato il valore in DN dell'errore di quantizzazione su tutti gli elementi della matrice di partenza, normalizzando il dato di errore massimo al valore bianco (255), ai fini di amplificarne la visibilità.

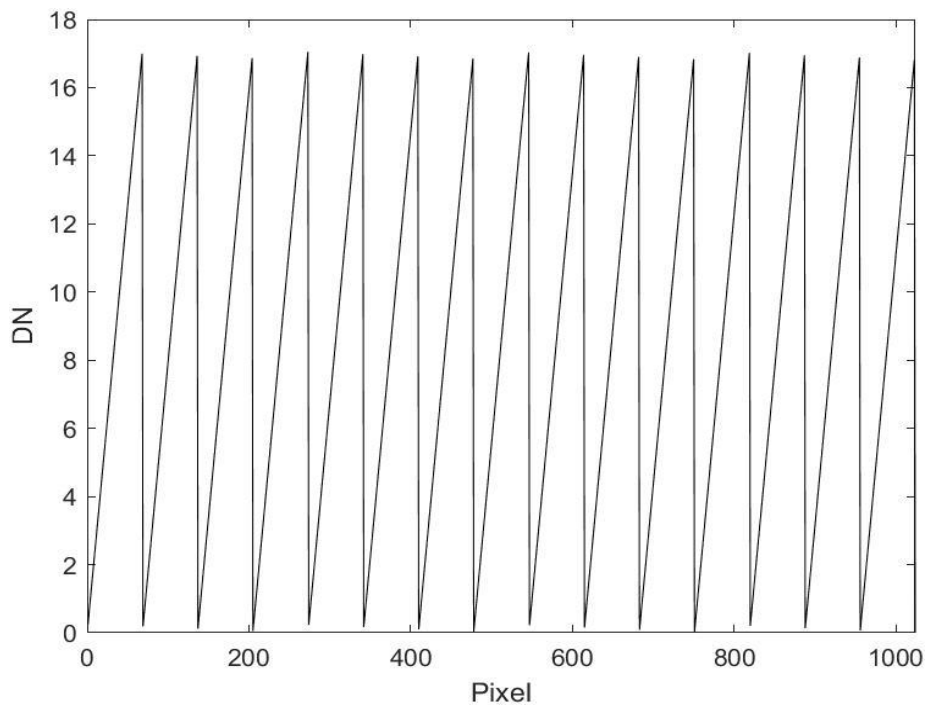


Figura 39 Errore di quantizzazione nel caso a 4 bit per una riga, espresso in DN

Anche nel caso dei 4 bit l'andamento dell'errore di quantizzazione per una singola riga (Figura 39) presenta dei valori di massimo pari al LSB: infatti, nel caso di immagine a gradiente (e in ogni immagine in generale) il valore dell'errore di quantizzazione non può mai essere superiore al valore assunto dal LSB.

## 4. CONCLUSIONI

All'interno dell'elaborato di tesi è stata inizialmente descritta una tipica architettura di camera, comprendente del sistema ottico di lenti e specchi, e dei sensori fotoelettrici.

Si è analizzata la struttura di questi, facendo particolare attenzione alle differenze che si articolano tra le tipologie di sensori CCD e CMOS riguardanti il processo di generazione dei fotoelettroni e della loro conversione in segnale elettrico, evidenziando i vari vantaggi e/o svantaggi che caratterizzano i due modelli.

Successivamente sono state introdotte le possibili fonti di rumore dovute all'utilizzo dei CCD e CMOS, e ne è stata presentata una descrizione di come queste provochino degli effetti negativi sull'output finale del processo.

Si è cercato di ottenere, per la maggior parte dei rumori, un'immagine contenente gli errori presi in considerazione, affiancata dalla rappresentazione dello stesso rumore ricavata mediante PANGU.

La visualizzazione degli errori per via grafica rappresenta infatti la scelta migliore per favorire una rapida comprensione degli effetti e dell'entità dei rumori trattati, resa immediata dall'utilizzo di un'unica immagine pulita di partenza alla quale sono stati applicati tutti gli effetti singolarmente.

Del fenomeno di Dark Current e Shot Noise, si è studiato nel dettaglio l'aspetto teorico riguardante la generazione di questo contributo involontario di elettroni: per l'applicazione pratica si è introdotta il rumore termico di White Noise, visto che lo Shot Noise di Dark Current condivide la stessa natura termica.

Per il problema del Blooming si è posta particolare attenzione nella simulazione dell'errore, la quale ha confermato i risultati ottenuti attraverso PANGU.

Le simulazioni del Fixed Pattern Noise hanno sottolineato una differenza evidente tra l'immagine affetta da rumore e quella pulita, quindi si è analizzata la tecnica di Flat Fielding in modo da attenuare gli effetti negativi dell'errore.

Il Readout Smear Noise è stato descritto indicando la differenza tra le tipologie Full Frame e Frame Transfer Imager dell'array del sensore, che sono caratterizzate da due modalità di lettura differenti, e quindi da due errori di entità diversi: la simulazione del rumore è stata svolta considerando l'applicazione Full Frame.

## BIBLIOGRAFIA

- [1] J. Janesick. (2001). *Scientific Charged-Coupled Devices*. SPIE Press, Bellingham, Washington, USA.
- [2] G. C. Holst (1998). *CCD Arrays, Cameras and Display*. SPIE Optical Engineering Press, Bellingham, Washington, USA.
- [3] S. Murchie, A. Mick, L. Prockter, A. Rivkin. (2017). *MDIS CDR/RDR Software Interface Specification*. JHU/APL, PDS Geosciences Node, Washington University.
- [4] M. Konnik, J. Welsh. (2014). *High-level numerical simulations of noise in CCD and CMOS photosensors: review and tutorial*. Faculty of Engineering and Built Environment, University of Newcastle, Australia
- [5] Christian, J. A. (2010). *Optical Navigation for a Spacecraft in a Planetary System*. The University of Texas at Austin.
- [6] S.E. Schröder, T. Maue, P. Gutiérrez Marqués, S. Mottola, K.M. Aye, H. Sierks, H.U. Keller, A. Nathues. (2017) *In-Flight Calibration of the Draw Framing Camera*. Icarus
- [7] J. Janesick. (2007). *Photon Transfer*. SPIE Press, Bellingham, Washington, USA.
- [8] H. Ren, T. T. Hu, Y. L. Song, H. Sun, B. C. Liu, M. H. Gao. (2018). *An improved electronic image motion compensation (IMC) method of aerial Full-Frame-Type area array CCD camera based on the CCD Multiphase structure and hardware implementation*. MDPI, Basel, Switzerland.
- [9] James R. Wertz, Wiley J. Larson. (2008) *Space Mission Analysis and Design*. Microcosm Press (CA) e Springer (NY)
- [10] Fossum, E. R., Hondongwa, D. B. (2014). *A Review of the Pinned Photodiode for CCD and CMOS Image Sensors*. IEEE Journal of the Electron Devices Society, Vol 2, No 3.
- [11] Fossum, E. R. (1997). *CMOS Image Sensors: Electronic Camera-On-A-Chip*. IEEE TOED, Vol 44, No 10.
- [12] B. T. Wysocki, M. A. Marciniak. (2008). *Discrimination between electronic and optical blooming in an InSb focal-plane array under high-intensity excitation*. ELSEVIER, Infrared Physics and Technology.
- [13] M. Dunstan, I. Martin. (2019). *Planet and Asteroid Natural Scene Generation Utility (PANGU) User Manual*. Space Technology Centre, University of Dundee, Scotland, UK.
- [14] ATMEL® Corporation. (2003). *Area Array CCD Image Sensor (1024x1024 pixels): TH7888A*.

- [15] Y. Jauregui-Sanchez, P. Clemente, P. Latorre-Carmona, E. Tajahuerce, J. Lancis. (2018). *Signal-to-noise of single-pixel cameras based on photodiodes*. INIT, Universitat Jaume I, Castellò.
- [16] P. Martinez, A. Klotz. (1998) *A Practical Guide to CCD Astronomy*. Cambridge University Press, Cambridge, UK

## SITOGRAFIA

- [1] PDS: Planetary Plasma Interaction, MESSENGER archive page, Mercury Dual Imaging System, Planetary: <https://pds.nasa.gov/>.
- [2] Smear in CCD-Sensors, PCO Computer optics:  
[https://www.pco.de/fileadmin/user\\_upload/pco-publications/pco\\_pub\\_20000107\\_smear\\_e.pdf](https://www.pco.de/fileadmin/user_upload/pco-publications/pco_pub_20000107_smear_e.pdf)

## APPENDICE

Si riportano di seguito le funzioni in linguaggio MATLAB® generate durante lo svolgimento dell'elaborato per simulare gli effetti dei rumori analizzati.

---

```
function [B] = Dark_Current_Shot_Noise(A,T,Ap,It,Dfm)
%Funzione che riceve in ingrasso la matrice corrispondente all'immagine
priva di rumori (A), la temperatura operativa in Kelvin (T), l'area del
pixel in µm (Ap), il tempo d'integrazione in secondi (It) ed il valore
della figura di merito di dark current (Dfm). Restituisce in output la
matrice corrispondente all'immagine affetta da Dark Current Shot Noise.
Un valore iniziale di Dfm che può essere utilizzato è 0.5

A=A(:, :, 1); %considerando un'immagine di ingresso a tre canali
Trm=300;
q=1.6e-19;
k=8.617332478*10^-5;

Eg=1.557-((7.021*10^-4)*T^2)/(1108+T);
C=Dfm*Ap/((q*Trm^1.5)*exp(-(Eg/(2*k*Trm))));
Jdark=(C*T^1.5)*10^(-(Eg/(2*k*T)));
Ndark=Jdark*Ap*It/q;
shot_dark=sqrt(Ndark);

size1=size(A);
B=zeros(size1);
med=0;
r=poissrnd(shot_dark,2048);
for i=1:size1
    for j=1:size1
        B(i,j)=A(i,j)+r(i,j);
    end
end

figure(1)
imshow(A)

figure(2)
imshow(B,[0 255])

end
```

---

```
function [D,F] = Blooming(C,threshold,sigma,e,gauss_flag)
%Funzione che riceve in input la matrice che rappresenta l'immagine
pulita (C), il valore di saturazione del pixel(threshold), il valore di
deviazione standard della gaussiana (sigma), il fattore di scala per
l'amplificazione dell'effetto blooming (e) ed una flag (1: on, 0: off)
per attivare l'effetto dato dalla gaussiana (gauss_flag). Restituisce in
output la maschera contenente solo l'effetto del rumore (D) e la matrice
finale rappresentante l'immagine affetta dal rumore (F)
```



```

syms y

C=C(:, :, 1);
size1=(sigma*4)-1;
filtersize=2*size1+1;
fprintf('Dimensione del filtro: %d\n',filtersize);
[row,col]=size(C);
D=zeros(row+2*size1,col+2*size1);
[n,m]=size(D);
scalefactor=(2*pi*sigma^2)*255;

for i=1:row
    for j=1:col
        if C(i,j)>=threshold
            mu_x=i-1;
            mu_y=j-1;
            f=@(x,y) scalefactor*(1/(2*pi*sigma^2))*exp(-0.5*((x-
mu_x)/sigma)^2+((y-mu_y)/sigma)^2));

            for k=(i-size1+1)*e:(i+size1-1)*e
                for h=j-size1+1:j+size1-1
                    if gauss_flag==1
                        D(k+size1+1,h+size1+1)=max(D(k+size1+1,h+size1+1),f(k,h));
                    end
                    D(k+size1+1,j+size1)=255;
                end
            end
        end
    end
end

D=D(size1+1:size1+row,size1+1:size1+col);
for i=1:row
    for j=1:col
        F(i,j)=max(C(i,j),D(i,j));
    end
end

figure(1)
hold on
title('Maschera del rumore da applicare')
imshow(D,[0 255])

figure(2)
hold on
title('Immagine con rumore')
imshow(F,[0 255])

end

```

---

```

function [raw,S,PN] = Fixed_Pattern_Noise(K)

```

```
%Funzione che riceve in ingresso la matrice corrispondente all'immagine
priva di rumori (K). In output vengono restituite tre matrici: (raw)
l'immagine contenente il rumore FPN, (S) la matrice dell'immagine
ottenuta dopo la calibrazione Flat Fielding e (PN) i coefficienti di non
uniformità dei pixel
```

```
K=K(:,:,1);
[row,col]=size(K);
A=zeros(row);
r=normrnd(1,0.07,row);
raw=zeros(row);
for i=1:row
    for j=1:col
        A(i,j)=K(i,j);
        A1(i,j)=A(i,j)*15;
        NU(i,j)=A1(i,j)*r(i,j);
        raw(i,j)=(NU(i,j)/15);
        flat(i,j)=1/r(i,j);
    end
end

figure(1)
imshow(raw,[0 255])

shot_flat=rand(1);
shot_raw=rand(1);
read_noise=rand(1);
mu=mean(mean(flat));
for i=1:row
    for j=1:col
        S(i,j)=mu*(raw(i,j)*flat(i,j));
varianza(i,j)=((raw(i,j)/flat(i,j))*shot_flat)^2+shot_raw^2+read_noise^2;
        sigma_S(i,j)=sqrt(varianza(i,j));
        PN(i,j)=sigma_S(i,j)/raw(i,j);
    end
end

figure(2)
imshow(S,[0 255])

end
```

---

```
function [B,F] = Readout_Smear_Noise(A,t_exp,t_col,k)
%Effetto del readout smear noise: in ingresso la matrice rappresentante
l'immagine (A), il tempo di esposizione espresso in secondi (t_exp), il
tempo di trasferimento di una singola colonna espresso in secondi (t_col)
e il coefficiente per scalare l'effetto (k). In output la maschera
contenente il rumore da applicare all'immagine pulita (B) e l'immagine
affetta dal rumore (F)

[row,col]=size(A);

B=zeros(row);
A2=zeros(row);
```

```

A1=A/t_exp;

for i=1:col
    A2(:,i)=A1(:,i)*(t_col*k);
    B(:,i+1:col)=B(:,i+1:col)+A2(:,i);
end

for i=1:row
    for j=1:col
        F(i,j)=A(i,j)+B(i,j);
    end
end

figure(1)
hold on
title('Maschera del rumore da applicare')
imshow(B,[0 255])

figure(2)
hold on
title('Immagine affetta da rumore')
imshow(F,[0 255])

end

```

---

```

function [A_analog,A_quant,A_quantDN,Error] =
ADC_quantization(Nwell,Nbit,gradient_value,N_pixel)
%Rappresentazione grafica del rumore provocato dalla conversione ADC a
partire da un gradiente utilizzato per visualizzarne l'effetto: i dati in
ingresso sono la capacità del pozzo potenziale espressa in numero di
fotoelettroni (Nwell), il numero di bit utilizzati per quantizzare il
segnale (Nbit), il valore massimo del gradiente espresso in percentuale
del valore bianco (gradient_value) ed il numero di pixel del lato
dell'immagine (N_pixel) in output. Gli output sono la rappresentazione
del segnale iniziale (A_analog), del segnale quantizzato (A_quant),
dell'associazione ai valori DN (A_quantDN) e dell'errore dovuto alla
quantizzazione (Error)

NDN=Nwell/15;
A_analog=zeros(N_pixel);
[row,col]=size(A_analog);
N_ADC=NDN/N_pixel;
val2=0;
for i=1:col
    val2=val2+gradient_value*NDN/N_pixel;
    A_analog(:,i)=val2;
end

ADC_code=2^Nbit-1
A_quant=floor(A_analog*ADC_code/NDN);
A_quantDN=A_quant*(NDN/ADC_code);
Error=A_analog-A_quantDN;

figure(1)

```

```
imshow(A_analog,[0 255])
figure(2)
imshow(A_quant,[0 255])
figure(3)
imshow(A_quantDN,[0 255])

figure(4)
subplot(121)
hold on
title('Intensità del segnale iniziale')
plot(1:N_pixel/10,A_analog(1,1:N_pixel/10),'b-')
ylabel('DN')
xlabel('Pixel')
subplot(122)
hold on
title('Intensità del segnale quantizzato')
plot(1:N_pixel,A_quantDN(1,1:N_pixel),'r-')
ylabel('DN')
xlabel('Pixel')

figure(5)
plot(1:N_pixel,Error(1,1:N_pixel),'k-')
xlim([0 N_pixel]);
ylabel('DN')
xlabel('Pixel')

figure(6)
imshow(Error,[0 max(max(Error))])
end
```

## **RINGRAZIAMENTI**

In primo luogo, voglio ringraziare il Professor Paolo Tortora per avermi permesso di approfondire la sua materia ed aver alimentato la mia grande passione per l'ambito spaziale.

Mi preme particolarmente ringraziare anche il mio correlatore Francesco Fiori, che mi ha seguito durante tutto lo svolgimento dell'elaborato con pazienza e disponibilità.

Infine, ringrazio la mia famiglia per avermi supportato durante tutto il percorso accademico, insieme a chi mi è stato accanto durante questi anni.