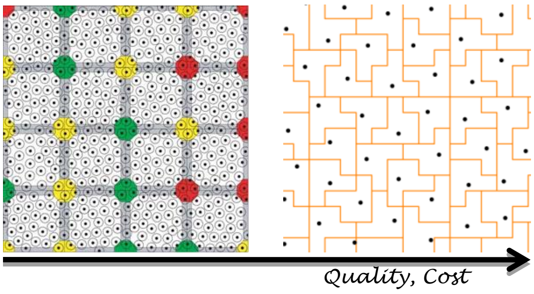
Noise Compression with AA Patterns

Abdalla G. M. Ahmed
University of Konstanz

Many sampling applications in graphics need uniform but irregular point sets; A.K.A "blue noise".
Such point sets are difficult to produce (expensive), but also difficult to pack (store) for reuse.

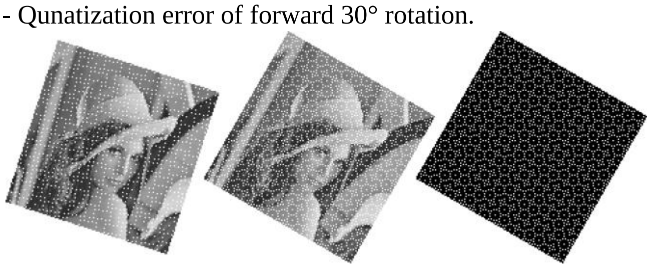
Classic Packing: Tiling



AA Patterns

Origin

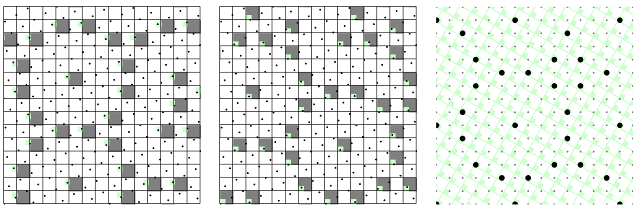
- Qunatization error of forward 30° rotation.



- Quantization error of certain transformations

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \left\lfloor \frac{1}{2} \begin{bmatrix} \alpha & -1 \\ 1 & \alpha \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \right\rfloor; \quad 1 < \alpha < 2$$

Development



Formulation

$$\mathrm{AA}(\alpha) = \{(x, y) : x, y \in \mathbb{Z}; dX < t; dY < t\},$$

where

$$dX = \mathrm{dX}(x, y) \triangleq \{(\alpha x + y)/2\}$$
$$dY = \mathrm{dY}(x, y) \triangleq \{(\alpha y + x)/2\},$$

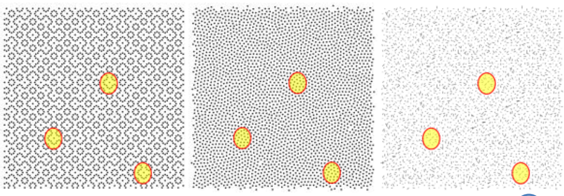
and

$$t = (\alpha - 1)/2.$$

Indexes (dX, dY) — Threshold (t)

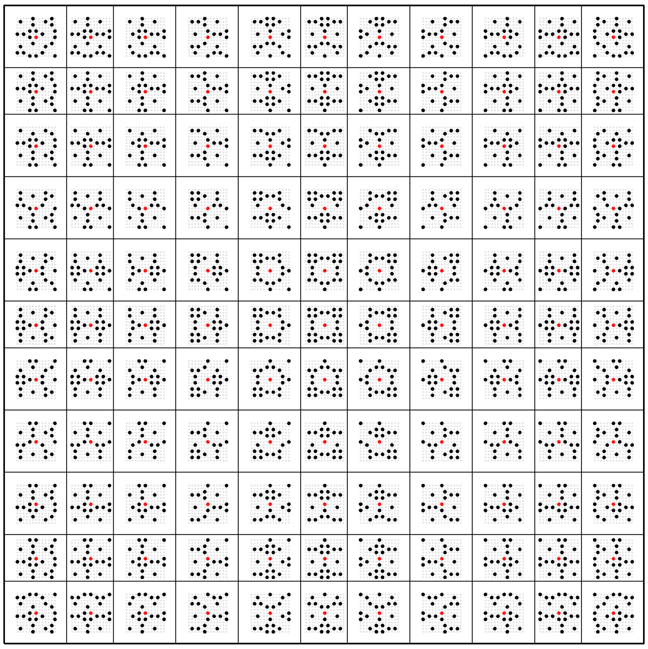
Morphing Concept

1. Take an AA Pattern,
2. Optimize it to match a target profile,
3. Record the displacements.



Similar neighborhoods lead to similar displacements

Neighborhood Maps



Data Sheet

- Packing time: 1s (Lloyd) - 3m (target-matching)
- Pack size: 64kB is sufficient
- compact self-contained unpacking function
- Unpacking:100M points per second
- Different noises (blue, green, etc.): yes
- Random access: yes
- Adaptive: not yet

Actual Code

```
For each grid point (x, y)
    dX = frac((alpha * x + y)/2)
    dY = frac((alpha * y + x)/2)
    if (dX < t) and (dY < t) {
        row = binarySearch(T, dY)
        col = binarySearch(T, dX)
        output (x, y) + D[row, col]
    }
```