CS 563 Advanced Topics in Computer Graphics
Sampling Techniques

by Wadii Bellamine

Outline

- Why do we need sampling techniques?
- What are the characteristics of good sampling techniques?
- What are some common techniques and how are they implemented ?
  - Random Sampling
  - Jittered Sampling
  - N-Rooks Sampling
  - Multi-jittered sampling
  - Hammersley sampling
- Results

Why do we need sampling?

1. To avoid aliasing when sampling pixels
2. Cameras with a finite-area lens are used to show depth of field. The lens must therefore be sampled.
3. To represent area lights and soft shadows more accurately, we must sample the light surfaces.
4. Global illumination and glossy reflection requires sampling of BRDFs and BTDFs.

Regular Pixel Sampling Example

- Problem 1: A form of aliasing – “jaggies”

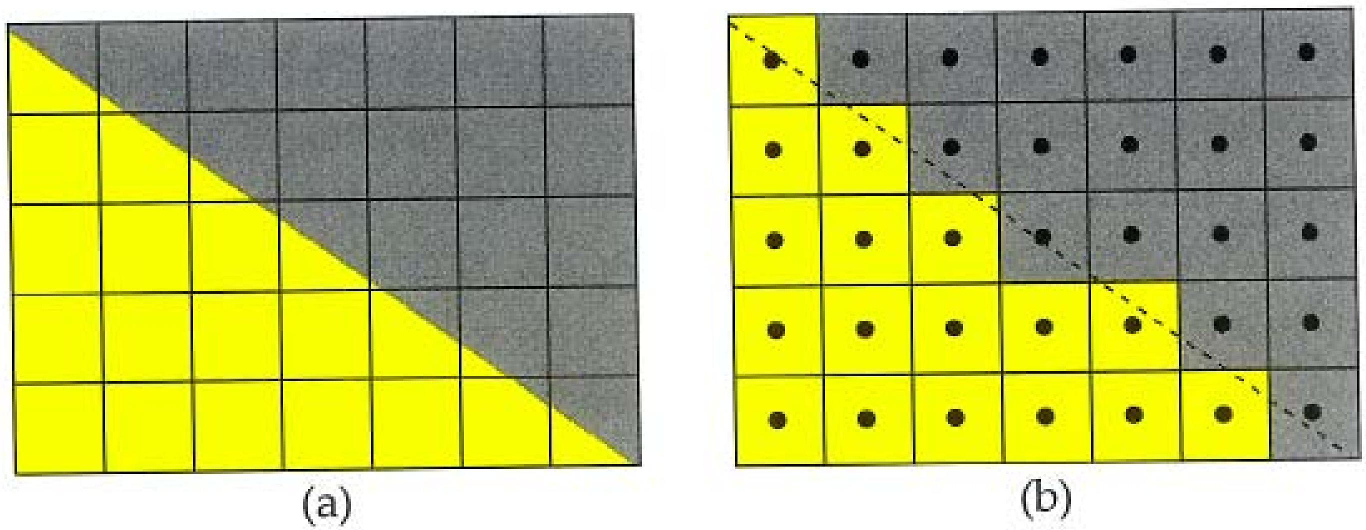


Figure 4.1. (a) Sharp edge; (b) ray traced at 7 × 5 pixels with rays through the black dots.

Regular Pixel Sampling Example

- Brute force solution: Increase image resolution

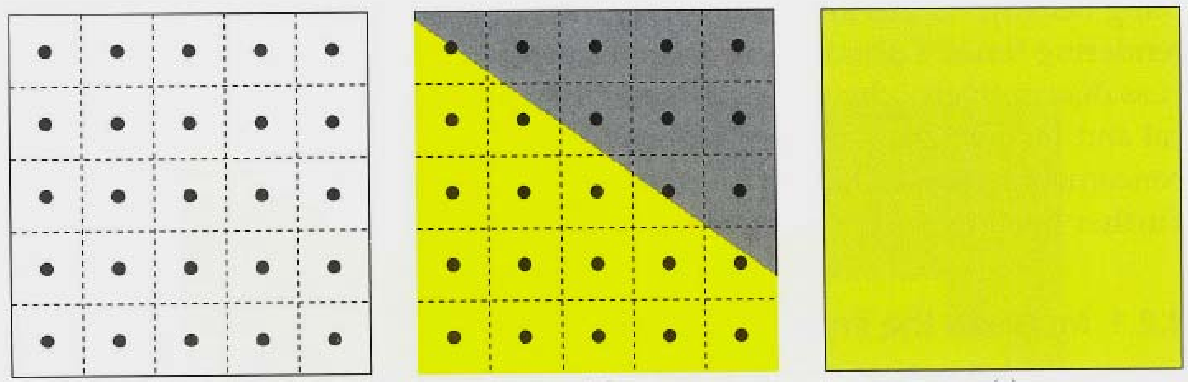
  - We still have the jagged edges

  + However, the eye will not perceive them at high enough resolutions

  - Requires a lot more processing

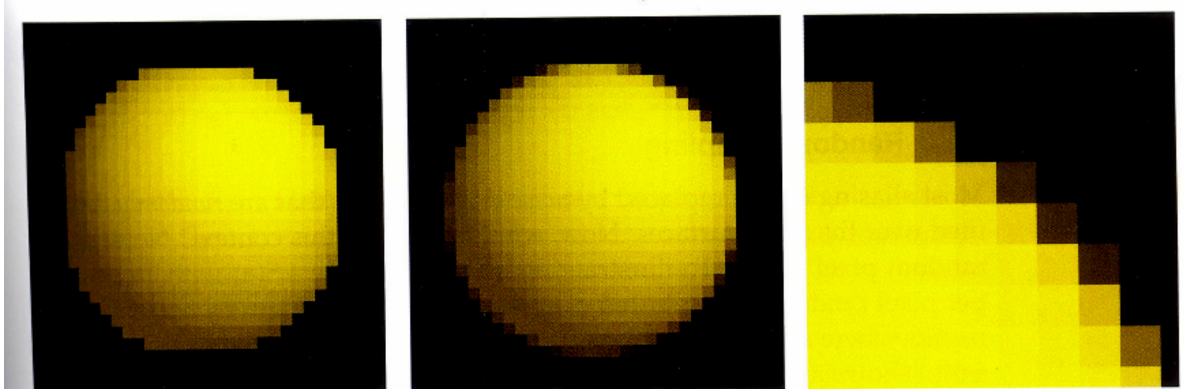
Regular Pixel Sampling Example

- Sampling solution:



(a) (b) (c)

Figure 4.3. (a) Regular sampling on a 5 × 5 grid in a single pixel; (b) regular sampling of the top-left pixel in Figure 4.1; (c) rendered color of top-left pixel.

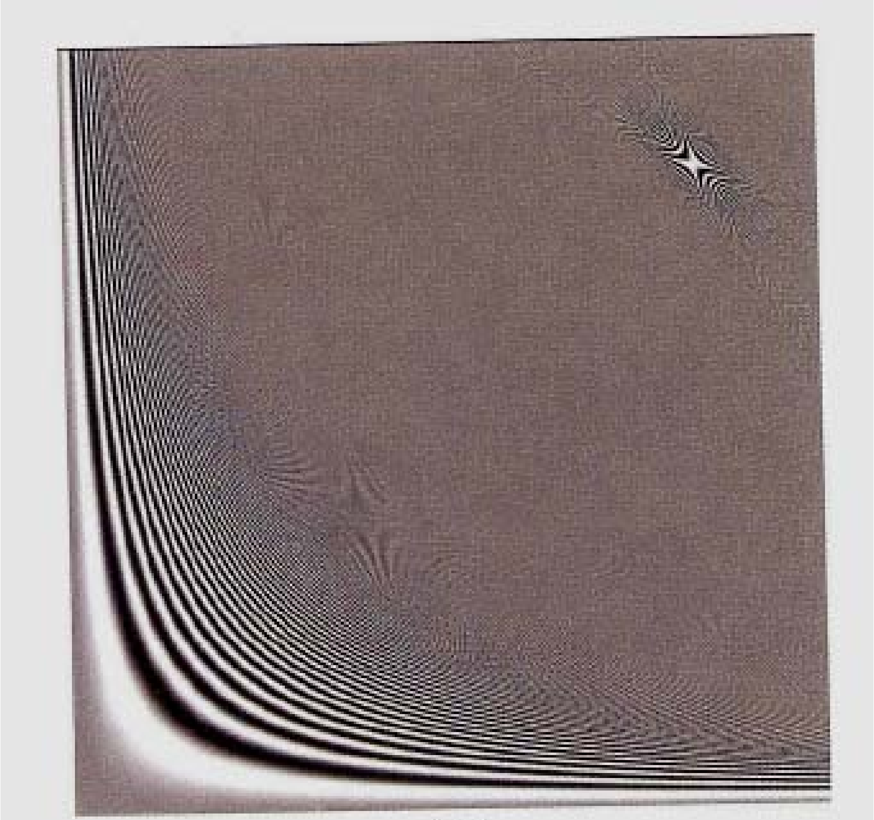


(a) (b) (c)

Figure 4.4. Shaded sphere: (a) one sample per pixel; (b) 16 samples per pixel; (c) enlarged view of top-right section of (b).

Regular Pixel Sampling Example

- "Regular" because rays are cast through the center of each grid cell. No randomization or shuffling was introduced. The equal spacing of the samples is said to be regular.
- Problem: Does not eliminate other aliasing artifacts such as moiré patterns:



(a)

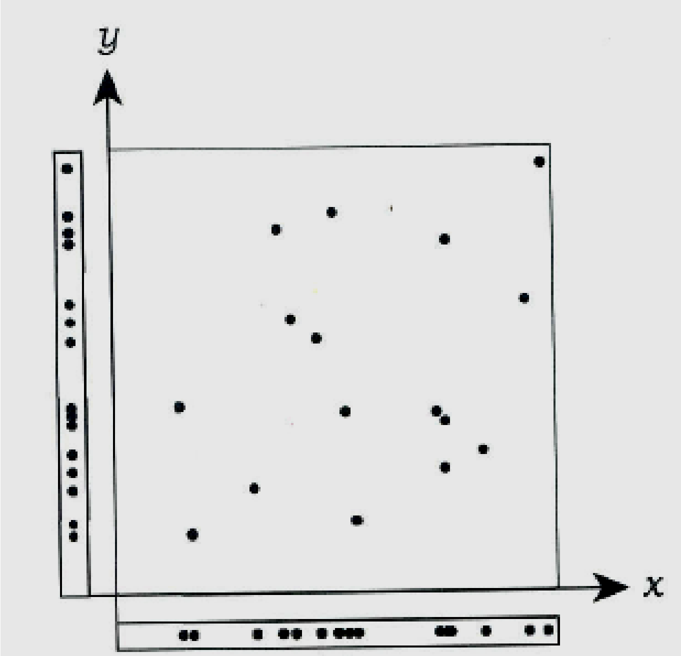
Cause?

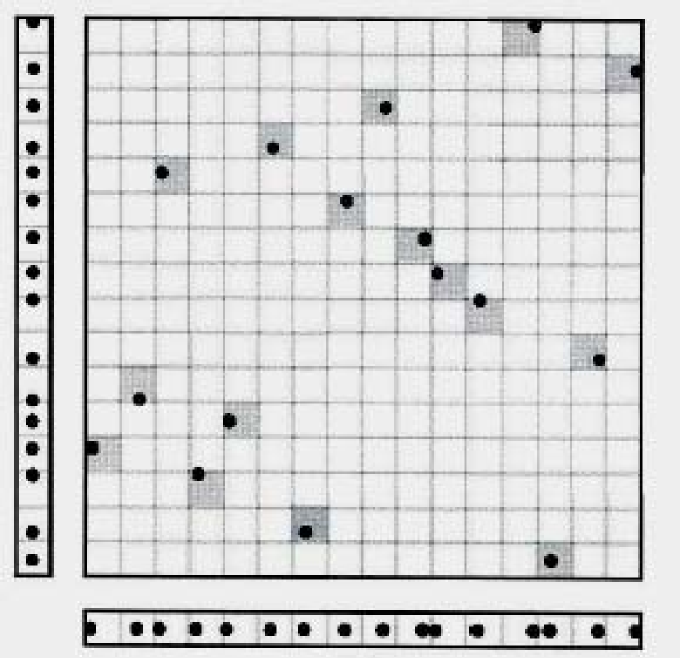
Solution?

Problem with solution?

Characteristics of good sampling

1. Samples are uniformly distributed over the unit square: reduces clumping and gaps
2. 1D x and y projections of unit square are also uniformly distributed





Random Sampling vs. 16 - Rooks

Characteristics of good sampling

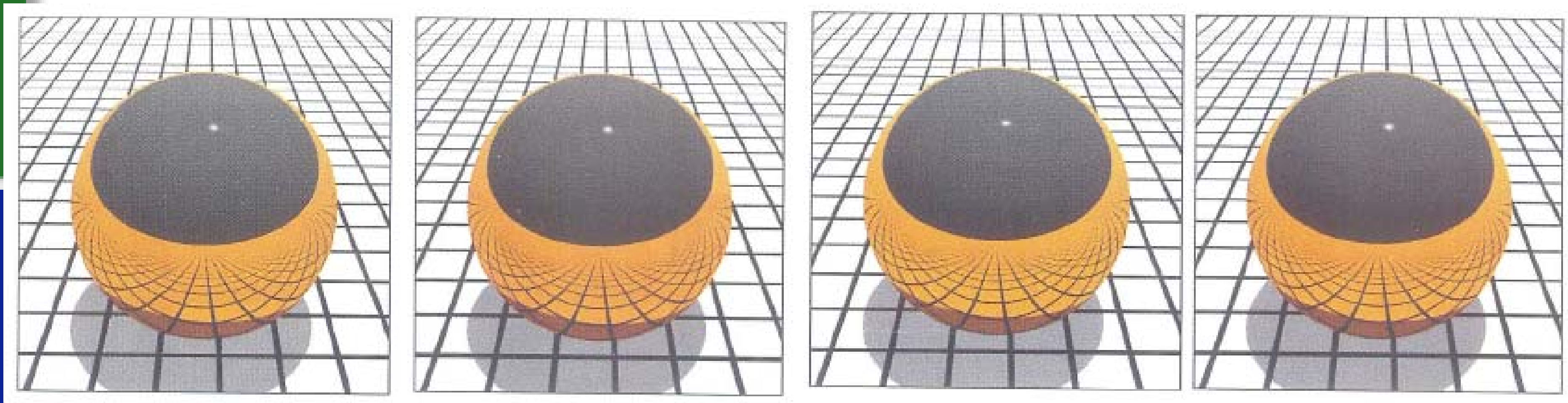
3. Some minimum distance is maintained between samples points.

Samples with the above three characteristics are called *well-distributed* (Hammersley samples are).

4. Do not want regular spacing, where the distance between sample points in the x and y directions is constant (as seen in slide 7).

If we just use a large number of samples, we can barely see the difference between the sampling techniques used.

Characteristics of good sampling



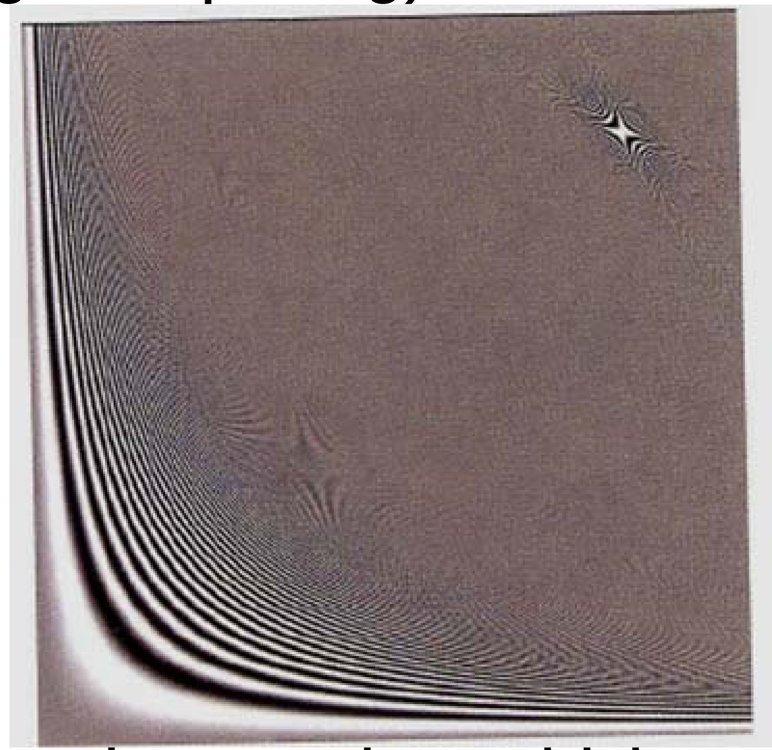
Q: So why do we care about all these different techniques?

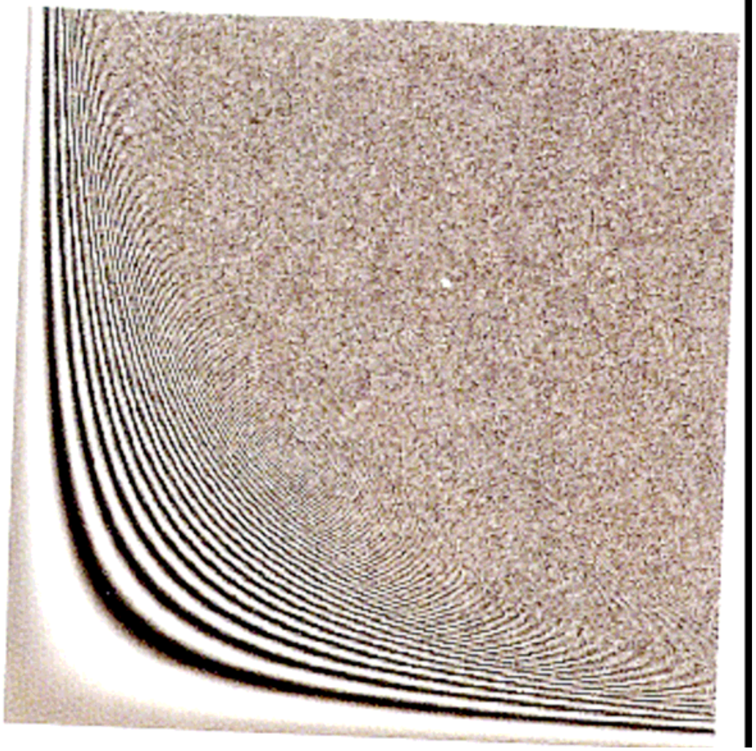
A: Because increasing the number of samples reduces efficiency, we must choose a number small enough to produce satisfactory results.

The best technique for a given application can require three times less samples than the worst technique.

Sampling Patterns: Random Sampling

- Simply randomly distribute the samples over the pixel (add a random value to the x and y of each sample in a regular spacing):





- Introduces noise, which can be unpleasant at low resolutions

Sampling Patterns: Random Sampling

- Fails the first three characteristics of good sampling:

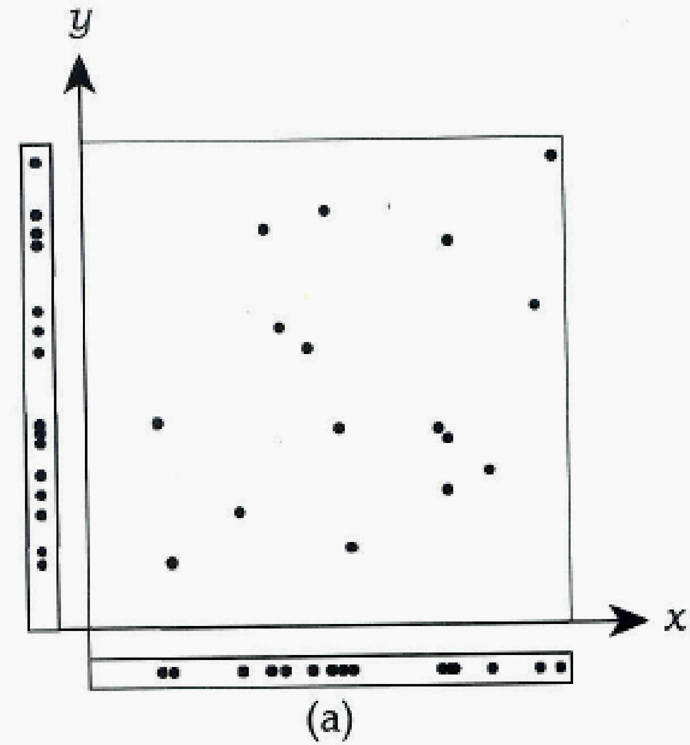


Figure 5.8. (a) 16 random samples with x- and y-projections; (b) 256 random samples.

Sampling Patterns: Jittered Sampling

- Choose a number of samples that is a perfect square, call this n^2
- Divide each pixel into an n x n grid.
- Randomly place a single sample within each cell of this grid.
- Each pixel is said to be stratified, and the domain is the n x n grid.
- Each cell is called a stratum, where strata are independent , but together cover the entire domain (i.e. no gaps or overlaps).

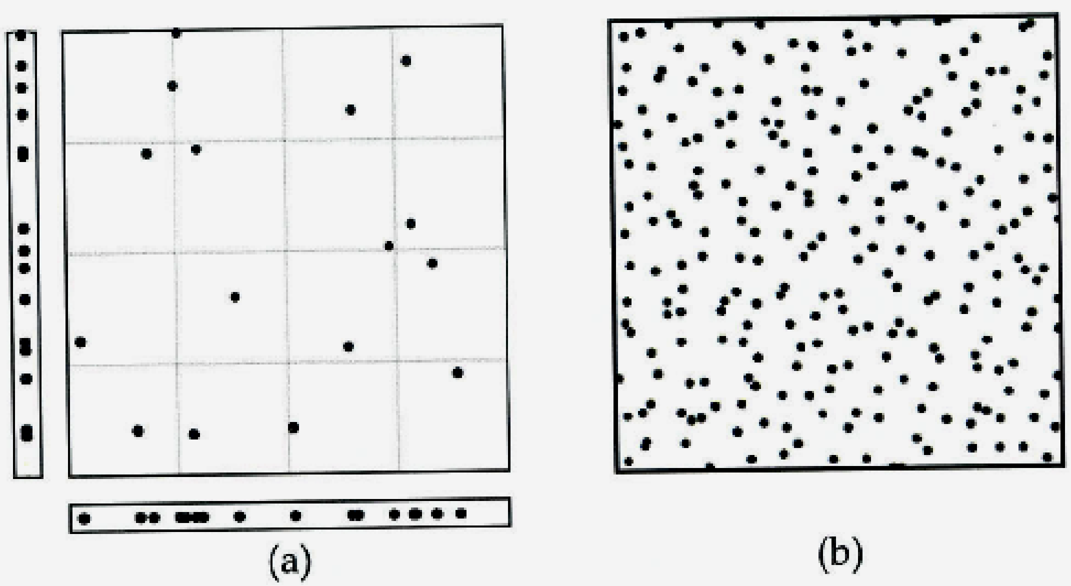


Figure 5.9. (a) 16 jittered samples with x- and y-projections; (b) 256 jittered samples.

Sampling Patterns: Jittered Sampling

Q: How can we modify the code for regular sampling to easily implement Jittered sampling ?

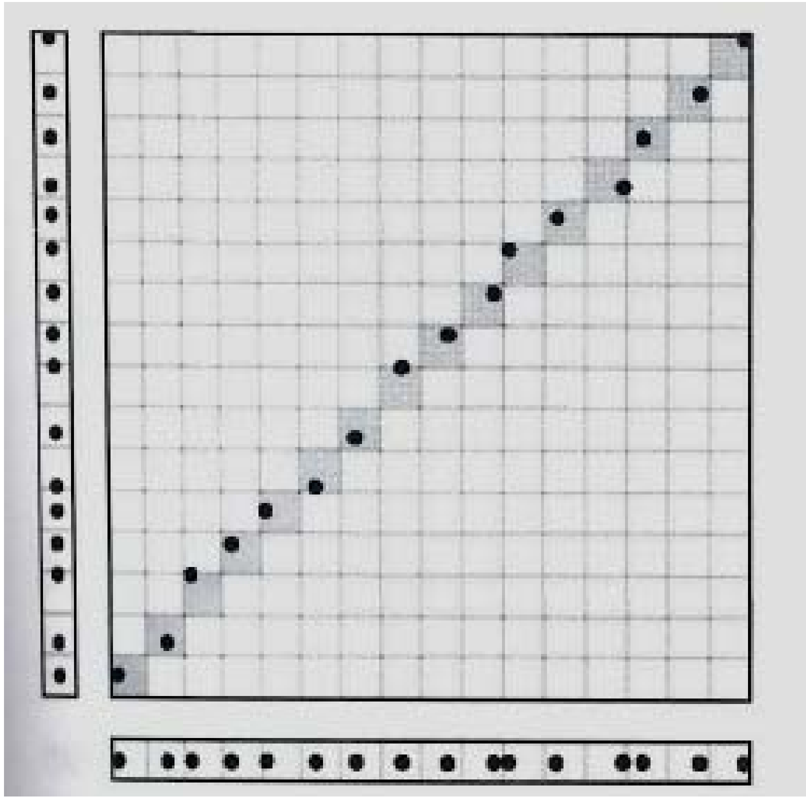
Q: What is the problem with this technique ?

Sampling Patterns: n-Rooks Sampling

- Developed by Shirley in 1991
- Achieves more uniform distributions in the 1D projections.
- n samples are placed in an n x n grid, such that there is a single sample, or rook, in each row and each column.

Sampling Patterns: n-Rooks Sampling

- Step 1: Randomly place one sample in each cell along the main diagonal of the grid:

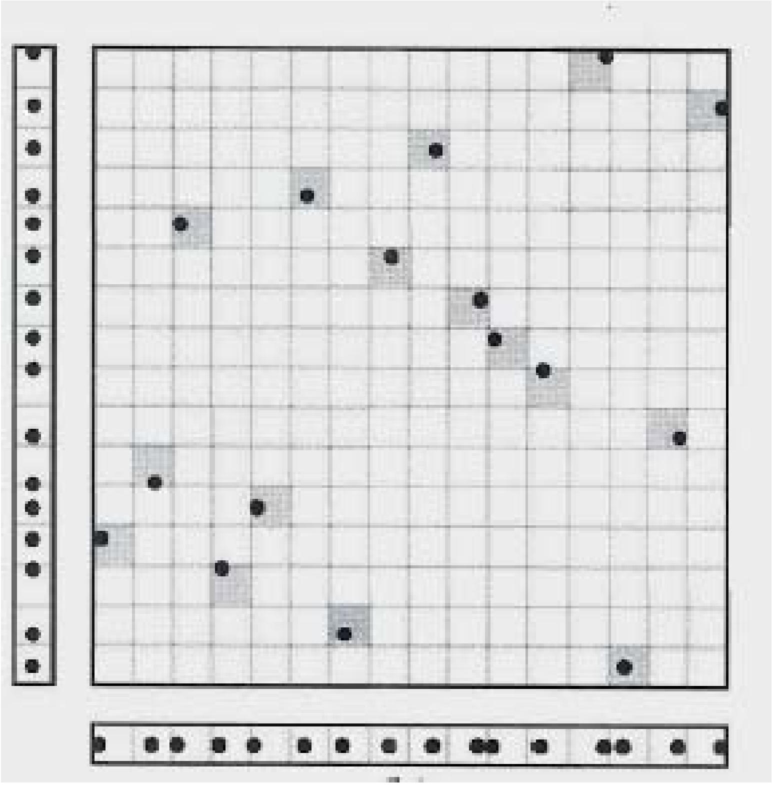


Sampling Patterns: n-Rooks Sampling

```
void
NRooks::generate_samples(void) {

    // generate samples along main diagonal

    for (int p = 0; p < num_sets; p++)
        for (int j = 0; j < num_samples; j++) {
            Point2D pv;
            pv.x = (j + rand_float()) / num_samples;
            pv.y = (j + rand_float()) / num_samples;
            samples.push_back(pv);
        }

    shuffle_x_coordinates();
    shuffle_y_coordinates();

}
```

Sampling Patterns: n-Rooks Sampling

- Step 2: Randomly shuffle the x and y coordinates while maintaining the n-rooks condition:

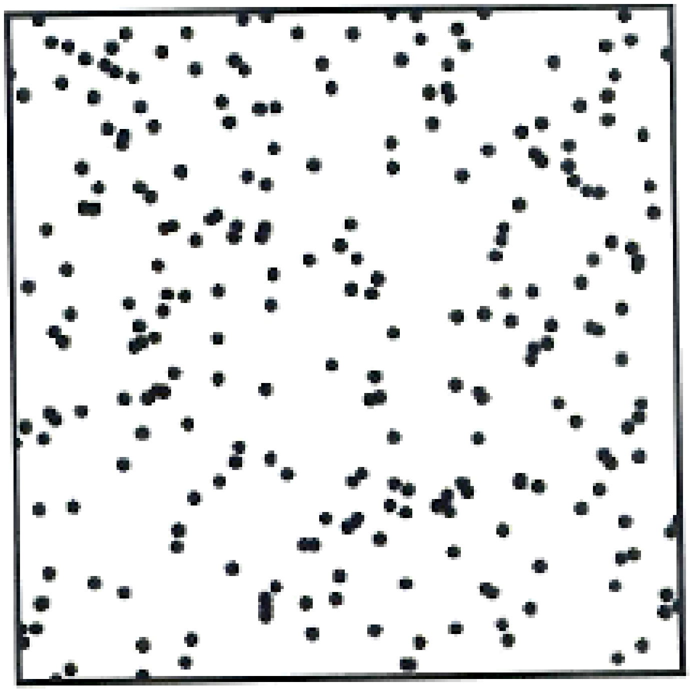


Sampling Patterns: n-Rooks Sampling

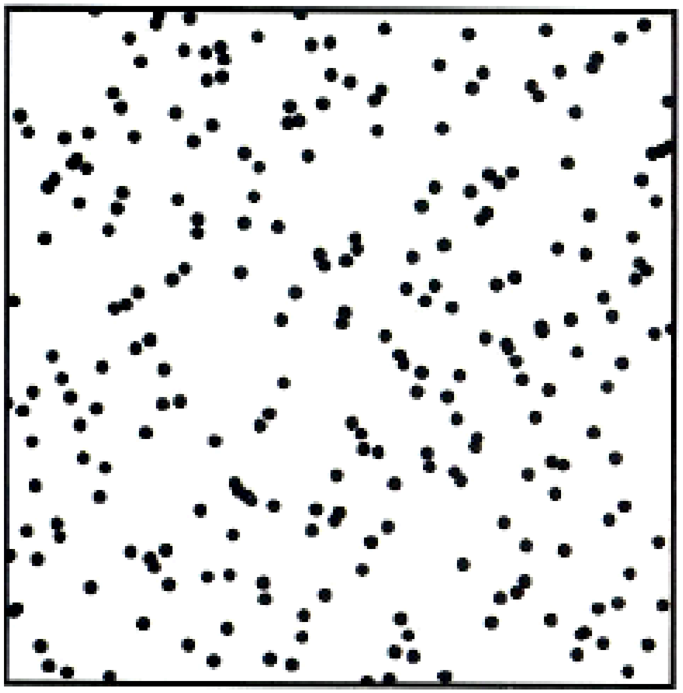
```
void
NRooks::shuffle_x_coordinates(void) {
    for (int p = 0; p < num_sets; p++)
        for (int i = 0; i <  num_samples - 1; i++) {
            int target = rand_int() % num_samples + p * num_samples;
            float temp = samples[i + p * num_samples + 1].x;
            samples[i + p * num_samples + 1].x = samples[target].x;
            samples[target].x = temp;


}
```

Sampling Patterns: n-Rooks Sampling

- Problem: We fixed the 1D projections, but ruined the 2D distribution! It is no better than that of random sampling:



Random



N-Rooks

Sampling Patterns: Multi–Jittered sampling

- Developed by Chui et al. in 1994 to fix 2D distribution of n-Rooks sampling by combining n-Rooks and Jittered sampling.
- Uses a two-level grid: We have n samples, where n is a perfect square. The upper level grid is $\sqrt{n} \times \sqrt{n}$, and the subgrid is n x n.

Sampling Patterns: Multi–Jittered sampling

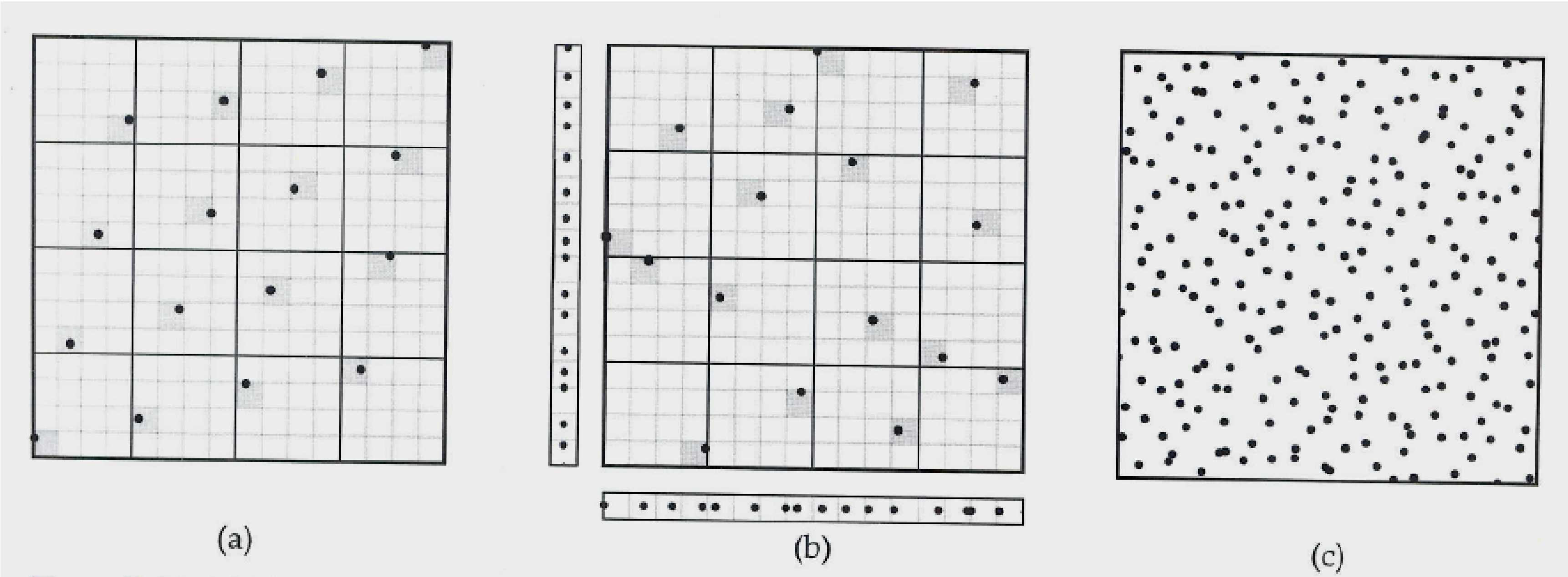


Figure 5.11. (a) 16 multi-jittered samples in the initial distribution; (b) after shuffling in the x- and y-directions; (c) 256 multi-jittered samples.

1. Systematically place a single sample in each cell of the upper-level grid while maintaining the n-Rooks condition.
2. Shuffle the samples in the sub-grid while still maintaining the n-Rooks condition and Jittered condition.

Sampling Patterns – Hammersley Sampling

- Developed by Hammersley and Hanscomb in 1964.
- Hammersley sampled are deterministic, not random, and are based on the representation of numbers in various prime bases.
- Binary representations give the best sampling distributions.

Sampling Patterns – Hammersley Sampling

- For an n x n unit square, the set of n Hammersley samples p_i is defined as follows

$$P_i = (x_i , y_i) = [1/n , \Phi_2 (i)]$$

Where Φ_2 (i) is the radical inverse function of the integer i base 2.

$$\Phi_2(i) = \sum_{j=0}^{n} a_j(i) 2^{-j-1} = a_0(i)\frac{1}{2} + a_1(i)\frac{1}{4} + a_2(i)\frac{1}{8} + \ldots .$$

- In other words, reflect the binary representation of i about the decimal point, and convert this to decimal representation.

Sampling Patterns – Hammersley Sampling

i			Reflection around the Decimal Point			$\Phi_2(i)$ (base 2)
1	=	1_2	$.1_2$	=	1/2	0.5
2	=	10_2	$.01_2$	=	1/4	0.25
3	=	11_2	$.11_2$	=	1/2 + 1/4	0.75
4	=	100_2	$.001_2$	=	1/8	0.125
5	=	101_2	$.101_2$	=	1/2 + 1/8	0.635
6	=	110_2	$.011_2$	=	1/4 + 1/8	0.325
7	=	111_2	$.111_2$	=	1/2 + 1/4 + 1/8	0.875
8	=	1000_2	$.0001_2$	=	1/16	0.0625

Table 5.1. Binary representations and radical inverse functions for the integers 1–8.

Sampling Patterns – Hammersley Sampling

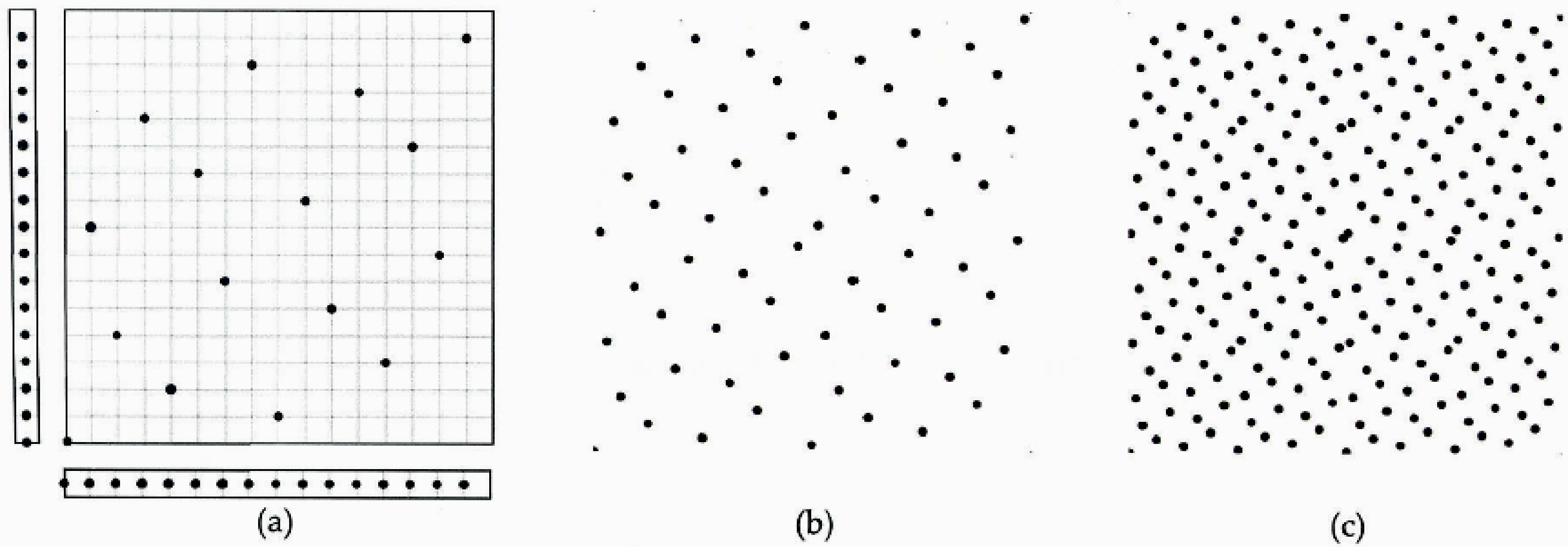


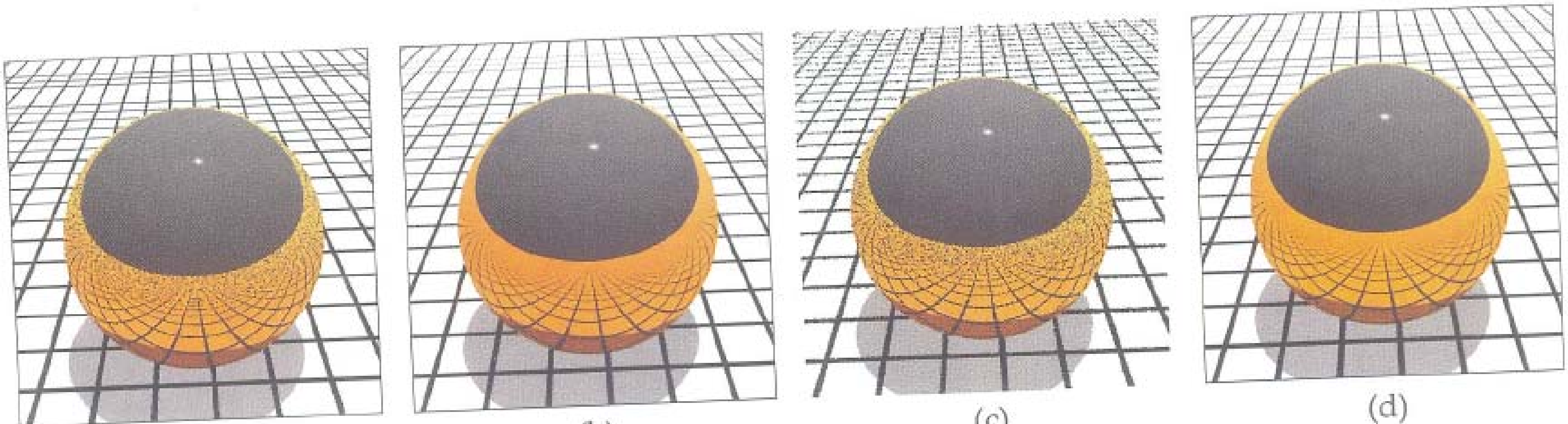
Figure 5.12. (a) 16 Hammersley samples with 1D projections; (b) 64 Hammersley samples; (c) 256 Hammersley samples.

- The 1D projections are regularly spaced – can lead to aliasing.

+ Points are well distributed in 2D, with a minimum distance between the points.

- Creates symmetries in the samples, which can also lead to aliasing.

Results



(a) (b) (c) (d)

Figure 5.19. Regular sampling with one sample per pixel (a) and 256 samples per pixel (b); random sampling with one sample per pixel (c) and 256 samples per pixel(d).

Results

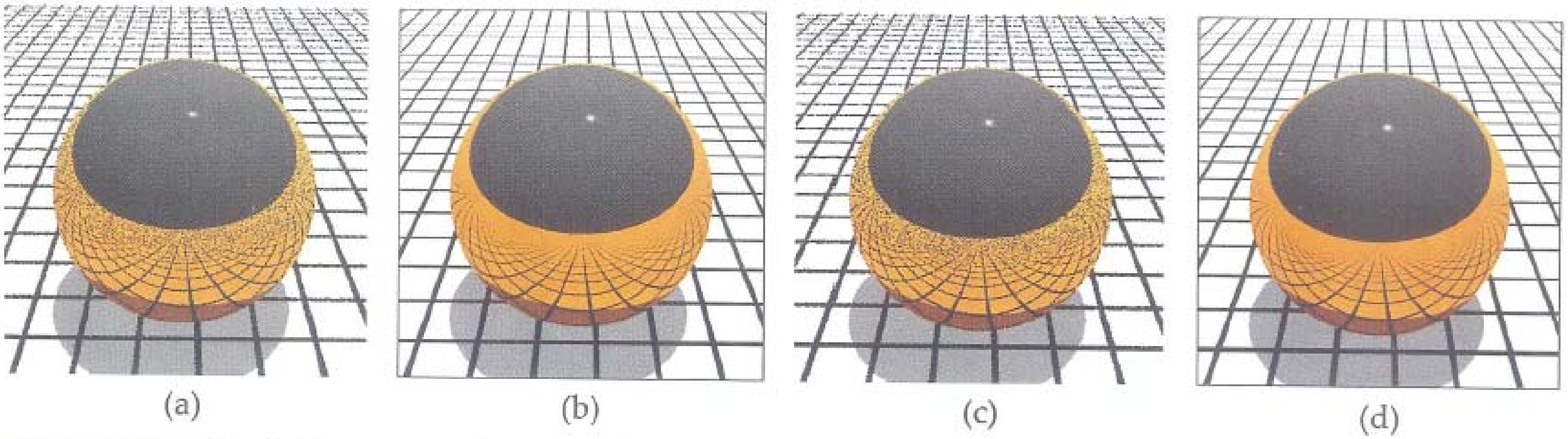


Figure 5.20. This is the same as Figure 5.19 but with jittered sampling in (a) and (b) and *n*-rooks sampling in (c) and (d).

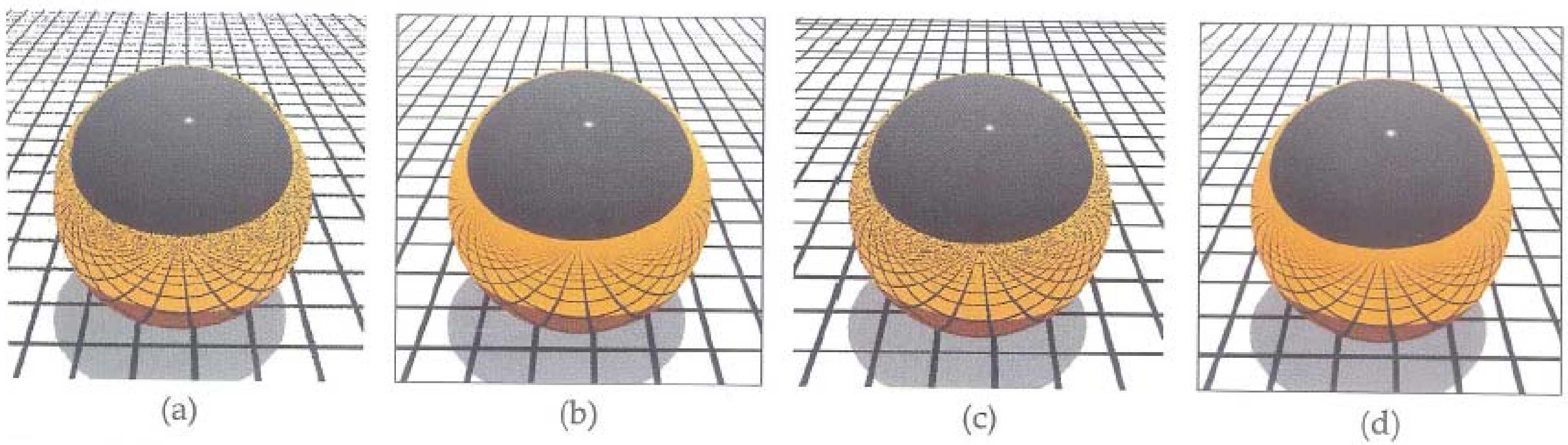


Figure 5.21. This is the same as Figure 5.19 but with multi-jittered sampling in (a) and (b) and Hammersley sampling in (c) and (d).

Reference

- All Images and information were taken from our textbook, Ray Tracing from the Ground Up.