2025-03-18-vglare(urp)

목표

veiling glare effect에 대해 efficient하고 ghost 개수에 scalability를 가진 rendering model 제안

기존 연구와 문제점

- 기존 방법에는 real-time과 higher quality 사이에 trade-off 발생(Ray tracing, Hullin et al, flare mapping based matrices)
- lens flare와 veiling glare를 분리하여 처리하면 성능 개선 가능

Contribution

- fourier series 기반 parametric 표현
    - glare model을 큰 주기를 갖는 주기 함수로 간주
    - screen size에 대응하는 high period > periodicity가 screen에서 보이지 않음
    - coefficient 계산을 효율적으로 계산하는 parametric 접근
- parametric multiple glares expression
    - screen의 모든 glare를 하나의 fourier series 틀에서 표현이 가능
    - moving light source and camera에 대한 coefficient를 look-up table > 효율적 계산
    - compute cost가 ghost의 개수 k에 independent, pixel 연산 단계에서 constant 유지
- Enhanced real-time glare rendering performance
    - glare rendering의 기존 접근보다 간소화된 연산으로 real-time rendering performance
    - Fourier series 기반 근사를 통해 glare quality 및 accuracy 개선
    - Physically-plausible veiling glare 모델을 효율적으로 표현

진행 사항

- Veiling glare는 ghosts가 만드는 단일 glare의 superposition으로 interpreted 됨
- glare model을 represent 할 수 있는 수학적 model 또는 function을 찾고 paper의 contribution에 적합한 parametric modeling이 가능한지 검토(fourier series + separability)

1d fourier series

Implementation process

1. fourier series coefficients의 적분 과정에서 analytic solution 가진 함수 선정
2. initial_update()에서 각 함수의 coefficients를 계산하여 메모리에 O(NK)시간으로 누적
3. render()에서 pixel 별로 전체 glare 값을 parametric하게 계산

함수 선정

- analytic integral value + bell-curve shape
- 목표로 하는 screen 제외 영역은 0으로 설정한다

$$f(x,y) = \begin{cases} 1 - x^2, & -1 \le x \le 1 \\ 0 & \text{otherwise} \end{cases}$$

general form($A - B(x-t)^2$), period = T

- coefficient

$$\begin{aligned} a_0 &= \frac{4A^3}{3BT} \\ a_n &= \frac{2}{T}\int_{-\frac{A}{B}+t}^{\frac{A}{B}+t} (A^2 - B^2(x-t)^2)\cos\left(n\frac{2\pi}{T}x\right)dx \\ &= \frac{8}{BT}\cos\left(\frac{2\pi n}{T}t\right)\left(\frac{\sin\left(\frac{2\pi nA}{BT}\right)}{\left(\frac{2\pi n}{BT}\right)^3} - \frac{A\cos\left(\frac{2\pi nA}{BT}\right)}{\left(\frac{2\pi n}{BT}\right)^2}\right), (n \ge 1) \\ b_n &= \frac{2}{T}\int_{-\frac{A}{B}+t}^{\frac{A}{B}+t} (A^2 - B^2(x-t)^2)\sin\left(n\frac{2\pi}{T}x\right)dx \\ &= \frac{8}{BT}\sin\left(\frac{2\pi n}{T}t\right)\left(\frac{\sin\left(\frac{2\pi nA}{BT}\right)}{\left(\frac{2\pi n}{BT}\right)^3} - \frac{A\cos\left(\frac{2\pi nA}{BT}\right)}{\left(\frac{2\pi n}{BT}\right)^2}\right), (n \ge 0) \end{aligned}$$

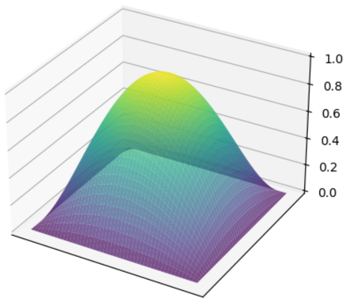
- fourier series

$$f(x) = \frac{4A^3}{3BT} + \sum_{n=1}^{N}\left[a_n\cos\left(\frac{2\pi n}{T}x\right) + b_n\sin\left(\frac{2\pi n}{T}x\right)\right]$$

- $O(N \cdot K)$ 만큼 각 series 항 n에 대하여 k개의 $a_{k,n}$ $b_{k,n}$ 를 누적하여 a_n,b_n 계산
- 누적 coefficients는 vector<float>(N) 형태로 메모리 저장
- 이후 각 pixel 별로 ghost의 개수 k에 상관 없이 $O(N)$으로 전체 glare 값을 계산, rendering

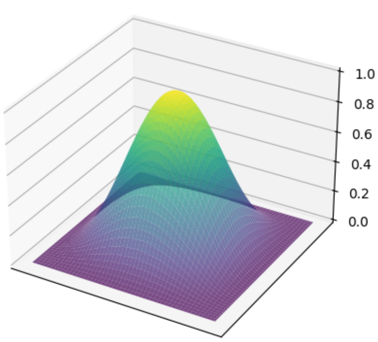
2d fourier series

$(1-x^2)(1-y^2)$



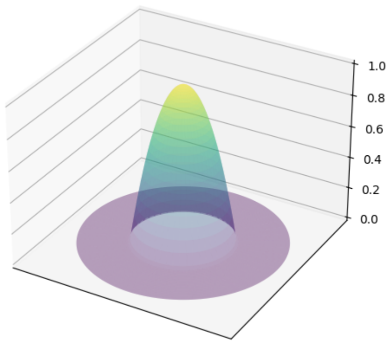
- 특징
    - 기존 1d fourier series의 glare model로 선정했던 $1-x^2$에서 확장
    - $\int\int f(x)f(y)dxdy$ 꼴이기 때문에 $\int f(x)dx \cdot \int f(y)dy$의 1d fourier series product
- 장점
    - 2개의 1d fourier series로 간주 > 2d extension에서 $\sin x \sin y$ 항 X
    - 구현이 쉽고 계산이 간단하다. 직관적이다.
- 단점
    - center 부근은 둥근 bell shape 이지만 z = 0의 floor가 square shape이기 때문에 glare model을 표현한다고 보기 어렵다.
    - cartesian coordinate에서 round shape floor > (x^2+y^2) 을 가지거나 circular해야 하고, boundary가 $z=0$ 또는 0에 가까운 함수를 찾아야 함.

$cos^2(x)cos^2(y)$



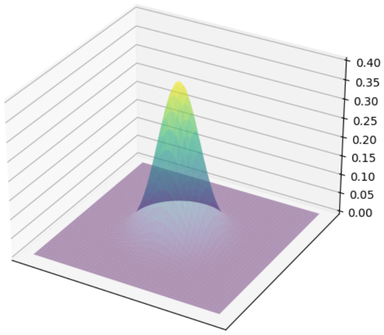
- 장점
    - 구현이 쉽고 직관적
    - 모든 구간에서 적분 가능
    - parametric veiling-glare paper의 2D extension 파트를 그대로 적용 가능
- 단점
    - $(1-x^2)(1-y^2)$보다 glare model을 잘 설명하지만 완벽한 bell-shape라고 보기 어렵다.

$1-(x^2+y^2)$



- 특징
  - round shape floor
  - 적분 구간을 $z \geq 0$인 $r \in (0, R)$, period = $2L$ 일 때, radius = L 인 원형 screen에서 근사
  - θ에 의존하지 않는 radial 2d fourier series > basis = bessel function $J_v(r)$
- 장점
  - 이론적으로 bell 모양을 정확히 구현
  - $\sum a_n J_0(r)$꼴이기 때문에 1d fourier series의 coefficient 계산과 비슷한 과정
  - glare가 이동하여 $\vec{r}-\vec{dr}$인 case에서도 $\vec{r}$과 $\vec{dr}$을 분리할 수 있는 separability
- 단점
  - bessel function $J_a(x)=\sum_{k=0}^{K}\frac{(-1)^k}{k!(k+a+1)!}\left(\frac{x}{2}\right)^{2k+a}$
    - x의 경우 real number이기 때문에 미리 table을 만들기는 어렵고 memoization까지는 쓸 수 있을 것 같다.
  - $J_0(||\vec{r}-\vec{dr}||)$ separability
    - $J_0(||\vec{r}-\vec{dr}||)=\sum_{m=0}^{M} a_k \kappa J_m\left(\frac{\alpha_k}{L}|\vec{r}|\right) J_m(\frac{\alpha_k}{L}|\vec{dr}|)\cos(m\theta)$
    - M 만큼의 시간이 추가 소요
    - colab(cpu 환경)에서 진행했을 때(N = 100, M = 100)의 경우 30분
  - 이론적으로 타당하지만 실제 구현에 어려움 존재, 추가적인 수학적 이해 필요

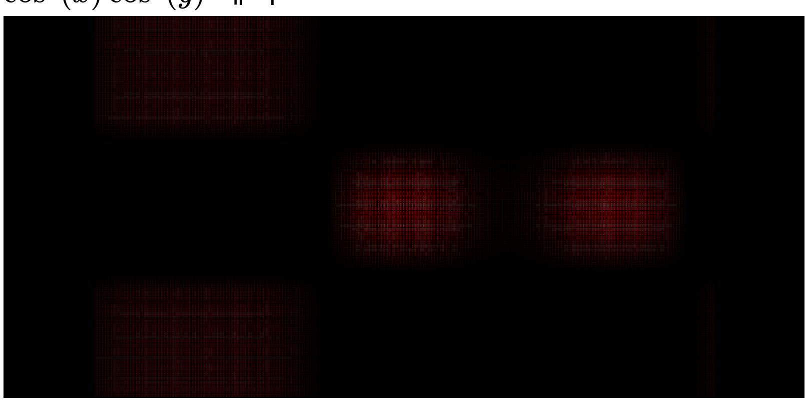
$\frac{1}{\sigma\sqrt{2\pi}}e^{-(x^2+y^2)/2\sigma^2}$



- 특징
  - 2d gaussian 꼴이기 때문에 cartesian 구간으로 적분해도 round floor shape
  - 이상적인 단일 glare model에 가장 가깝다.
- 장점
  - 모든 구간에서 미분 가능하기 때문에 수학적으로 쓰기 좋다
  - $e^{-(x^2+y^2)}$항을 $e^{-x^2} \cdot e^{-y^2}$꼴로 분해할 수 있고 1d fourier series의 곱 표현
  - parametric veiling-glare paper의 2D extension 파트를 그대로 적용 가능
- 단점
  - 적분 구간에서 발생하는 erf function을 다음과 같이 근사
    $erf(x) \approx 1 - \left(a_1\frac{1}{1+px} + a_2\frac{1}{(1+px)^2} \cdots + a_5\frac{1}{(1+px)^5}\right)e^{-x^2} + \epsilon(x), |\epsilon(x)| \leq 5 \times 10^{-7}$

추후 계획

- 적합한 glare model 후보군을 선택해 rex 상에서 구현
- 저장된 coefficients에 seires basis를 곱하고 누적, pixel마다 $O(1)$ scale로 값을 계산
- $\cos^2(x)\cos^2(y)$ 예시



- 두 개의 타원 꼴을 확인할 수 있다.