A NOVEL MOTION CONTROL WORKFLOW FOR CAPTURING AND REPRODUCING REALISTIC LENS FLARES

FILMAKADEMIE BADEN-WÜRTTEMBERG
animationsinstitut

CVMP2024

VINCENT MAURER
FILMAKADEMIE BADEN-WÜRTTEMBERG, ANIMATIONSINSTITUT

PROBLEM

Lens flares are caused by reflections and scattering within a camera system. While they can degrade image quality they are often used artisitcally.
Current image or simulation based techniques often fall short of recreating lens flares faithfully.

Fig 1: "The Deep Above" (C) Filmakademie Baden-Wuerttemberg 2024

RELATED WORK

Recreating lens flares in computer graphics is usually achived by either:

- Moving and blending images of lens flare ghosts [1]
- Hybrid 2D and raytracing techniques [2]
- Raytracing through virtual lenses [3]

They either rely on full knowledge about the optical system or need manual user recreation of a given lens. This is often not sufficient to recreate all lenses flare characteristics, like forward scattering and reflections of the mechanics.

NOVEL APPROACH

A camera attached to a motion control robot films a point light as a regular spaced grid. This creates a high density dataset of lens flare images.

The research compares traditional image interpolation to machine learning techniques like RIFE [4] and a custom CNN to recreate the flare patterns from the dataset.

METHOD

A state of the art cinema camera (Arri Alexa 35) and lens is mounted on a motion control robot. The robot rotates the camera around its nodal point in a serpentine pattern. With this method a set of different lenses with varying apertures is captured as a video (100k images per lens).

Fig 2: Capture Setup

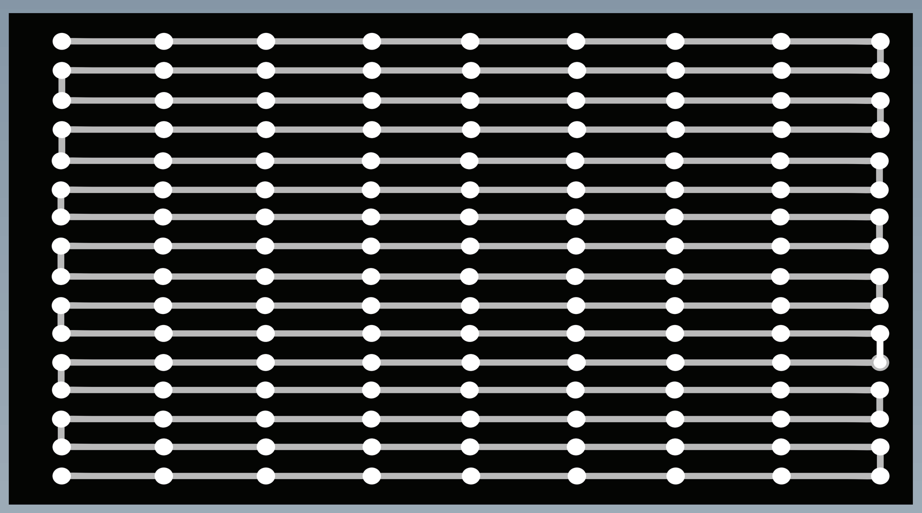

Fig 3: Serpentine Pattern (Simplyfied)

The dataset is denoised and stored as a high dynamic range image sequence that is converted into a spritesheet. A gizmo for the compositing application Nuke picks the closest images in the spritesheet and interpolates between them.

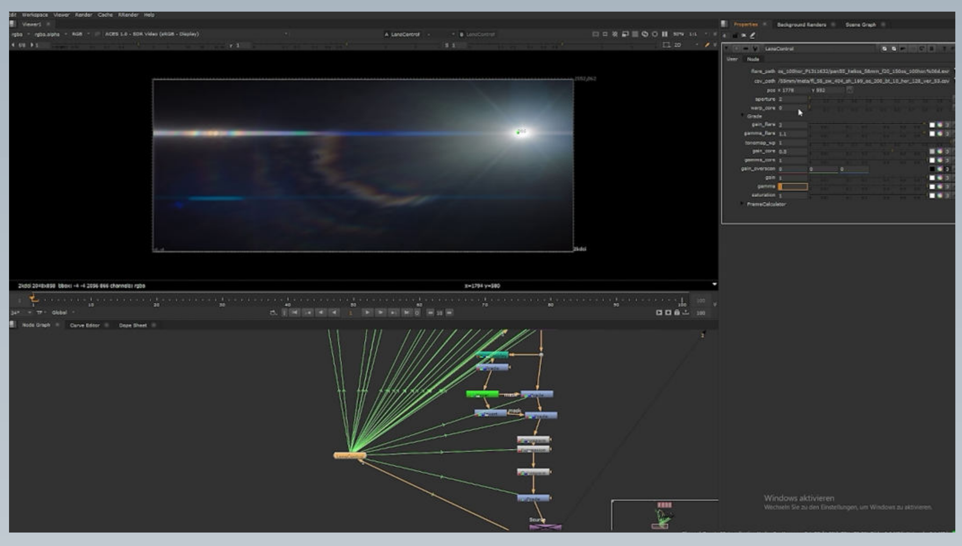

Fig 4: Nuke gizmo, lens flares "The Deep Above" (Fig1)

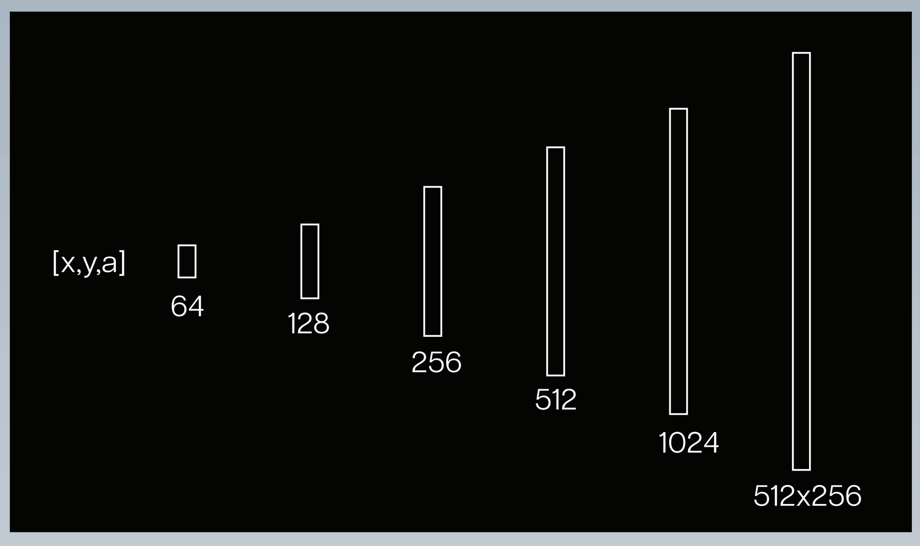

Fig 5: CNN Architecture

Additionally a CNN was trained on a vector of xy position and aperture with the corresponding flare image. The architecture is composed of 6 upsampling convolution layers, followed by a fully connected layer to the final resolution of 512x256. L2, L2 gradient and SSIM is used as a loss function. The CNN is implemented as a pytorch network and converted into Nuke's Cattery format [5] for easy user interaction.

RESULTS

The image based technique (Fig 6.1 + 6.2) shows good clarity in still images, but has noticable interpolation artifacts on the layered transparent ghosts. Its sprite sheet based method makes it suitable for use in realtime compositing and game engines (Fig 6.3).

Fig 6: Comparison

The machine learning techniques (Fig 6.4 - 6.6) show less clarity than the images based ones, but with the advantage of smoother interpolation between positions and apertures. Some interpolation issues might stem from a misalignment in the dataset. Therefore revising the capture method and exploring more advanced network archtiectures are an interesing avenue for further research.

Short paper, dataset and comparison video:
https://vincent-maurer.github.io/lens-flare-capture/

REFERENCES

[1] S. Jo, Y. Jeong, and S. Lee. Real-time nonlinear lens-flare rendering method based on look-up table. Journal of KIISE, 44:253–260, 03 2017. doi: 10.5626/JOK.2017.44.3.253.
[2] M. Hullin, E. Eisemann, H.-P. Seidel, and S. Lee. Physically-based real-time lens flare rendering. ACM Trans. Graph., 30(4), jul 2011. doi: 10.1145/2010324.1965003.
[3] E. Pekkarinen and M. Balzer. Physically based lens flare rendering in "the lego movie 2". In Proceedings of the 2019 Digital Production Symposium. ACM, 2019. doi: 10.1145/3329715.3338881.
[4] Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. Real-time intermediate flow estimation for video frame interpolation, 2022. URL https://arxiv.org/abs/2011.06294.
[5] https://community.foundry.com/cattery

Contact author: vincentmaurerlt@gmail.com