

Towards Spatial Airflow Interaction: Schlieren Imaging for Augmented Reality

Zhang Zhibin, Yuichi Hiroi, and Yuta Itoh, *Member, IEEE*

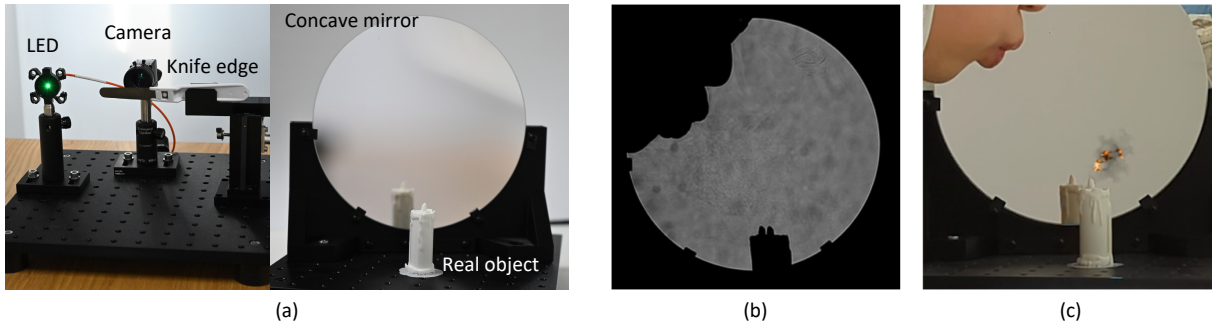


Figure 1: Our proof-of-concept system integrating Schlieren imaging to AR as spatial airflow interaction. (a) Our Schlieren imaging setup with a single-path reflective model. (b) A captured Schlieren image from our system when a person is breathing out within the detection area. The system extract directional information of the airflow including its pressure. (c) Video see-through AR view utilizing the spatial airflow information. The virtual candlelight rendered atop the real candle mockup is being swept away by the real breath. Please also refer to our supplementary demo video.

ABSTRACT

This work integrates Schlieren Imaging, a unique sensing modality, into Augmented Reality (AR) to explore ways to utilize invisible airflows for AR. Schlieren imaging is an imaging technique that visualizes the flow of fluids, which is normally invisible to the eyes. Theoretically, the technique can calculate the motion, pressure, temperature, and density of the airflow in our physical world. This unique, but less applied modality may expand interaction paradigms in AR and VR. We build a proof-of-concept AR system combined with Schlieren imaging that allows real airflow to affect virtual objects. The results of quantitative analyses show that our system can integrate different types of airflow with pressure values ranging from weak breathing actions to a heat gun up to 10m/s or 0.25m³/min airflow. We also showcase AR use cases including blowing out a virtual candle and a heat gun.

Keywords: Schlieren imaging, augmented reality, spatial airflow interaction

Index Terms: H.5.1 [Multimedia Information Systems]: Artificial, augmented, and virtual realities—; H.5.2 [User Interfaces]: Input devices and strategies—

1 INTRODUCTION

In augmented reality (AR) and virtual reality (VR), the research community explore interaction methods based on different modalities including natural interaction such as gestures, posture, and eye-tracking [19, 32].

Among such interaction modalities, a unique one is airflow interaction such breathing [7, 35]. Interestingly, breathing is the only bodily activity that directly affects the human heart and is conscious, and can also show information about our physical condition through AR devices for the treatment of hypertension, arrhythmia, etc. [28, 33].

Some works study breathing or blowing air as a direct input method. Existing blowing or breathing interaction methods are either based on common microphones or dedicated sensors [7]. The former is the most common options since microphones are commodity and small, making them easy to install in even head-mounted displays (HMDs) for VR and AR. The latter option is less common and their systems tend to be complicated and expensive.

Note that, microphones-based systems measure sound, thus the input is susceptible to interference of the environment noise and the information obtained is limited. In addition, the measured sound volume varies depending on the arrangement of the microphones and the loudness of the user's voice or environment sounds, which reduces reliability. Triggering is also another issue such as if a user is breathing or just speaking. More importantly, all these microphone-based works, in general, cannot directly measure spatial information of the sound waves or airflows by nature.

Schlieren photography is an optical method developed in experimental physics to visualize density changes in a volume of gas [4, 29]. The technique has enabled science and industry, especially in aeronautical engineering, to study the characteristics of airflow in transparent media [38, 39].

Using Schlieren measurement gives two major advantages over existing sensing schemes. Firstly, it can provide spatial information on air flows including speed and pressure. Secondly, airflow is measurable even when the flow makes no sound. Despite these useful properties, Schlieren imaging has limited application in VR and AR.

Therefore, this work aims to inspire the community with this unique sensing modality, and in particular, to demonstrate the concept of an AR system that takes advantage of the intrinsic usefulness of schlierens, namely spatial airflow.

We hope this work encourages the community to explore applying the Schlieren imaging modal to AR and VR applications.

Our major contributions include the following:

- proposing a concept of using Schlieren imaging for AR as a new interaction modal.
- developing a proof-of-concept AR design and system to demonstrate spatial airflow interaction.

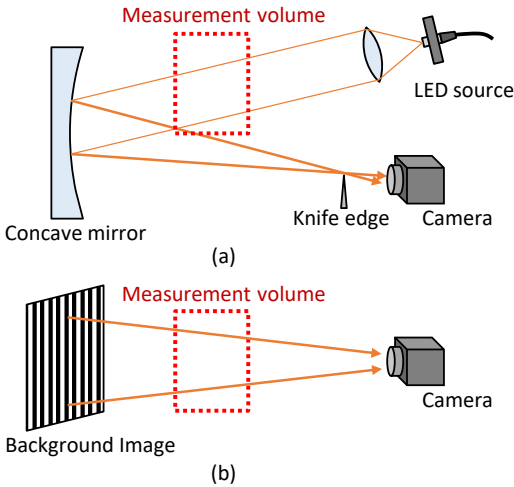


Figure 2: Two typical Schlieren imaging setups. Red dotted rectangles depict measurement areas. (a) A single-path reflective method, which is employed in our system. (b) Background-oriented Schlieren.

- providing tips and directions to apply Schlieren imaging in future AR/VR research.

2 RELATED WORK

We cover works on blowing and breathing interaction for AR/VR applications (Sec. 2.1), Schlieren measurement (Sec. 2.2), and particle image velocimetry for airflow’s velocity estimation (Sec. 2.3).

2.1 Blowing and Breathing Interaction

Blowing interactions use sensors such as microphones [15, 41], gas sensors [16], and depth cameras [8].

Microphones are probably the most common sensor in blowing interaction because they are inexpensive, easy to use, and compact. Microphones are also ubiquitous. Some works used those installed in laptops [27] and smartphones [11, 40]. Yeqing et al. developed an adaptive blowing interaction method based on a Siamese network by using a headset microphone to obtain the sound waveform of the blowing action [7].

However, since microphones can only measure sound waves, they are vulnerable to noise and provide limited information. For example, the measured volume will vary depending on where the microphone is placed and the volume of the user’s voice, which makes it difficult to do quantitative analyses of sound waves. Furthermore, for such applications, a system must distinguish whether the user is breathing or just speaking.

Unlike using microphones, Misha et al. implemented a blowing interaction system called breathVR using a commercial wearable biometric sensor (Zephyr BioHarness), to use breathing as a directly controlled input technique for VR games [35].

More importantly, all the above methods classify breathing actions but do not measure the properties of airflow. Classifying breathing actions into different types is useful for discrete input such as buttons. Yet these cannot be used as analog input or directly interact with virtual objects in an AR setup.

However, any methods above only detect human breathing actions, even though airflow contains more spatial information such as direction and pressure, which can be useful in AR where virtual contents interact with the real 3D world.

As a predecessor to our concept, Sawada et al. proposed BYU-BYU-View, a bidirectional telecommunication video system with an airflow interface [30]. Their airflow interface has an airflow input/output mechanism. The airflow input is an 8x8 two-dimensional

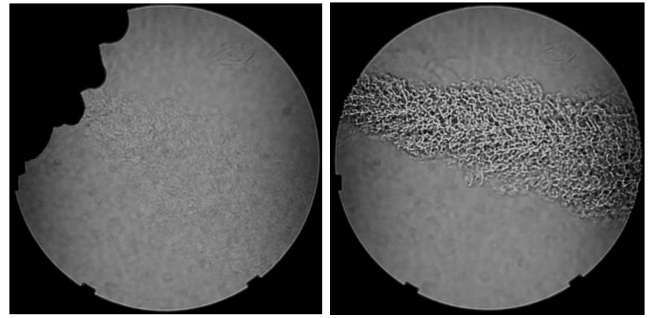


Figure 3: Schlieren images from our setup, which is using a single ray-path reflective model. The scenes are when (left) a person is breathing, and (right) using a hair dryer.

array of sensor modules that indirectly detect the input airflow speed by the tilt change of a free plate caused by the airflow. In addition, the airflow output is three rows of air blower arrays, one between each of the two rows of input arrays. Although their system has discrete spatial resolution, it can provide airflow intensity-based interactions such as user blowing on a table-sized space.

Compared to their input system, our system has a smaller tracking area, but can directly track the spatial behavior of actual airflow at a finer spatial scale thanks to the Schlieren imaging. The current implementation assumes a single airflow and calculates the direction, velocity, and pressure from the entire measured schlieren image every frame. This allows for more physically realistic interactions in AR applications.

2.2 Schlieren Measurement

Schlieren imaging and shadowgraphs convert invisible differences in the phase speed of light, i.e. refraction effect, into changes in intensity. The density changes appears as light or dark areas based on the Gladstone–Dale relation, a physics law that relates air density to its refraction states. The two methods were initially used for airflow visualization, then became a tool to quantitatively measure airflow along with the development of image processing [31].

Schlieren sensors typically have two implementation designs [14]. Fig. 2(a) is a typical reflective Schlieren setup which relates the gray intensity changes in Schlieren images to airflow’s refraction distribution using the knife-edge. As a variant, the rainbow Schlieren uses a rainbow filter instead of the knife edge to map the ray distortion direction into colors [1, 17]. Fig. 2(b) is a background-oriented Schlieren, which does not need other equipment but only one camera. When inhomogeneous media comes into a test volume, the pattern of the background images will be distorted. The method relies on the distortion of background to detect the hidden airflow [3, 26].

In general, the reflective Schlieren is optically more complex, yet can directly visualize the density change (Fig. 3). Whereas, the background-oriented Schlieren keeps measurement setup simpler while requiring an algorithm to extract the density change from the captured background. In our prototype, we employed the reflective Schlieren for the simpler computation cost.

Martínez et al. show that a classical Schlieren system can measure the temperature and velocity of the medium [23]. They adopted a Schlieren measurement method called the scaling Schlieren to calculate the temperature field of the airflow. This method helps build a relationship between the gray intensity changes in a Schlieren image and the property changes of airflow, which cannot only be used for temperature measurement but also for density and other properties of airflow. However, a single frame of the Schlieren image cannot determine the velocity of the fluid.

Particle image velocimetry (PIV) analyzes motion between con-

secutive Schlieren frames to obtain a tracked sequence and finally, velocity fields. This part will be covered in the next subsection.

Overall, Schlieren imaging can theoretically determine velocity, temperature, and many other properties of fluid flow that other sensor models cannot detect. Digital processing technology also enabled such quantitative analyses and the extraction of spatial information from 2D Schlieren images. This flexibility opened various applications in different fields [31]. Schlieren can measure properties including spatial flow vectors of airflow. Some works even rebuild the particles fluid in the air for modeling [25, 34].

In AR applications, virtual objects often interact with physical objects. We thus believe that AR applications can adopt Schlieren modal to expand this paradigm to invisible air medium compared to existing breathing interaction.

2.3 Particle Image Velocimetry

In AR or VR environments, objects are 3D and dynamic. We thus need to reconstruct temporal real-world airflows in 3D to use them in a virtual space. PIV is a powerful tool for reconstructions of airflow [20, 21]. Motion estimation of fluids is a challenging task and has important applications in research and industry. PIV gives instantaneous velocity measurements and related properties in fluids. PIV is used for non-intrusive measurement, quantitative analyses, and qualitative flow visualization [12].

PIV originally used a laser to illuminate the particle in the fluids and capture the images of the illuminated particle with a camera for measurement. Katrin et al. presented the first variational model that jointly solves sparse particle reconstruction and dense 3D fluid motion estimation in PIV large volumes [18]. Since PIV images with lasers and Schlieren images are all bright dots images, the PIV algorithm can be adopted for Schlieren measurement for velocimetry. Also, with the advance in digital technologies, real-time processing and applications of PIV became possible. There is already some excellent libraries and tools open source, such as PIVLab [37] and OpenPIV [22].

Adopting PIV's 3D fluid flow reconstruction pipeline directly into AR or VR interaction is not easy due to the computation cost. It is important that every action should have real-time responses when we want to design an interaction method. As we show in Sec. 3.2, to balance the complexity and the realism of measured airflow in our pipeline, we limit the airflow visualization as a single velocity particle for each given Schlieren image. We prove the concept by employing a video see-through scenario for demonstration.

3 METHOD

In the following, we describe the basic theory of analyzing the Schlieren images in Sec. 3.1. Then, in Sec. 3.2, we further elaborate our AR rendering and interaction pipeline based on a game engine. Note that another section (Sec. 4.2) covers the optical setup.

3.1 Schlieren Image Analysis via PIV

A Schlieren image contains properties of airflow implicitly. Assuming we have Schlieren imaging system (Sec. 4), we first elaborate on calculating the velocity and density via PIV as a spatial airflow interaction requires dynamic pressure of the airflow.

Generally, PIV needs seeding particles that are illuminated so that particles are visible from images. The analysis of velocity is actually measuring the motion of sequences of bright dots images. However, we can use this analysis algorithm in Schlieren measurement because Schlieren images are also sequences of bright dots images. The basic theory behind PIV calculation is the following. The algorithm needs two frames of Schlieren images to obtain the distance between two same particle fluids and divide by the time between two frames as the velocity of this particle fluid.

The algorithm finds the same particle fluids in this way. Firstly, we split the first image into many small windows. Then, we want to

find the displacement of these small windows in the second frame as a single vector, so we are looking for the most probable displacement of this image. We do an image processing pattern matching using a cross-correlation analysis. We look for the highest peak, which is the most probable position of the motion of the second image with respect to the first image. After we decide on one pixel for how far in the physical world and the time between two frames, we can calculate the velocity between two frames. For rendering and interactions, we have to calculate the average velocity of the whole airflow. The basic procedure behind this step is to draw a line from the airflow coming direction and add all vectors crossing this line on average.

Besides, pressure calculation requires airflow's density ρ . According to Gladstone-Dale's relationship [24], the relation between the airflow's density(ρ) and the refractive index(n) is following:

$$n - 1 = K_{GD}\rho \quad (1)$$

where K_{GD} is the Gladstone-Dale constant that depends on the gas composition and on the wavelength. The relation is basically showing a ray suffers a certain angle deviation when it passes through an inhomogeneous media. This angle depends on the refractive index and width of the media.

This equation shows that the airflow's density ρ varies directly as the refractive index n . Namely, the gray intensity changes of the Schlieren images vary directly as the airflow's density does. The airflow's density changes also mean the change in temperature, pressure, and many other properties. Following this physics theory behind Schlieren measurements, we can estimate the airflow's density from the gray intensity changes of the Schlieren images and can also calculate the temperature, and pressure. For this density estimation, we applied the scaling Schlieren methodn [23], which estimates the density from a Schlieren image by considering the base Schlieren image where no airflow exists [23]. For this offline calibration process, we measure intensity deviations versus knife-edge positions by adjusting the knife-edge position. This calibration result even further allows to estimate temperature distribution [2] (Sec. 4.3).

We also implemented an flow angle extraction algorithm to help with more flexible interaction under airflow types from one direction such as blowing, hair dryer, and spray. The algorithm does a linear regression after contour detection of the airflow. In our proof-of-concept system, one can move the direction up and down to interact with virtual objects. Although the robustness of this algorithm still needs to improve, this algorithm is currently enough to confirm our concept.

3.2 Rendering and Interactions

The previous section gives the spatial airflow direction and its force. These suffice to let real airflow interact with virtual contents. Yet, we also explore that visualization of the surrounding hidden airflow can also enhance the realism of the rendering in a virtual environment. Therefore, in this section, we briefly describe surrounding airflow rendering. In the virtual environment, we drew a basic texture of particle objects and adjust some initial properties. We designed our rendering system to change the simulation speed and the z-dimension of rotation of each particle, according to the calculated result from PIV analysis and angle detection.

To simulate the physical effects of the real world into the virtual environment, we added force, i.e. pressure, P to the visualized airflow based on the physics low:

$$P = v^2\rho/2, \quad (2)$$

where v and ρ are the airflow's speed and density, respectively.

Our current pipeline calculates the average velocity and density to obtain the pressure value. The basic procedure behind this step is

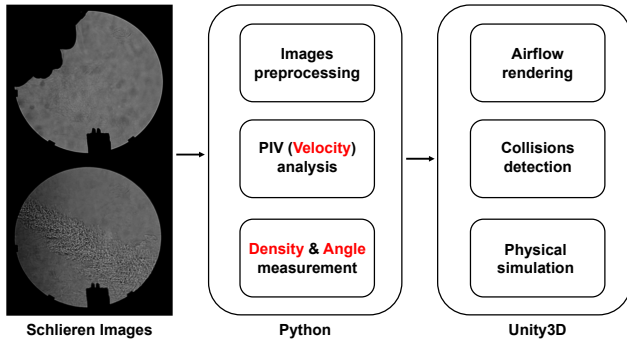


Figure 4: The pipeline of the whole system, including the Schlieren sensor, image analysis in python, rendering, and interactions in Unity3D. Schlieren images will be sent to python for analysis. The algorithm helps measure the velocity, density, and angle of the airflow then passes them to Unity3D for visualization and interactions.

to draw a line from the airflow coming direction and add all vectors crossing this line on average. We also count the mean of all gray intensity inside the contours of the airflow on average as the density we used in physical effects. We elaborate our software pipeline in Sec. 4.3.

4 TECHNICAL SETUP

In this section, we first introduce the pipeline of our system. Then we step into each part of the system to show their technical setup and output. We separate the whole system into three parts, a Schlieren system for detection, the algorithm for image analysis, and the game engine for rendering and interactions.

4.1 System Overview

Fig. 4 is the pipeline of the whole system. We first capture the airflow in the test area by the Schlieren optical system. The captured Schlieren images will send to python for analysis. After image preprocessing by OpenCV, we install OpenPIV to analyze every two frames of Schlieren images and output the velocity of the airflow. Then, we calculate the angle and pressure of the airflow by images processing. After finished analyzing, we send all results to Unity3D for rendering and physical effects. The particle system of Unity3D will help us render the airflow in the virtual environment. By setting the properties of the rendered particle, we can change its outlook, animation, and physical features. Once the particle detects colliders in the game scene, the game engine will help show physical effects, which achieve interactions with virtual objects.

4.2 Schlieren Setup

We employ a single ray-path reflective model as shown in Fig. 5. The red light path shows a ray that the camera does not capture because of the refraction caused by density changes in the test area. On the contrary, the blue line shows rays that can be captured.

As shown in Fig. 6, our hardware setup used a C-mount camera (Ximea MC031CG-SY, 2064 x 1544 pixels resolution, 122Hz) with a lens (Computar, 16mm), an LED light source (Thorlabs M530F2, 530 nm, 6.8 mW) coupled with a fiber patch cable (Thorlabs M38L01, $\varnothing 200 \mu\text{m}$, 0.39 NA), a concave mirror (750mm focal length, $\varnothing 16\text{cm}$), and a commodity utility knife for knife-edge.

Note that, the bigger mirror diameter expands areas the system can detect airflow, 16cm is enough for our airflow interaction. Another factor we should pay attention to is the focal length of the concave mirror. A longer path gives the rays that are refracted and diverted from the medium more time to diverge, thus a system will get better sensitivity at the cost of requiring larger space.

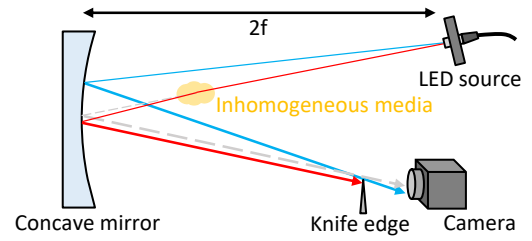


Figure 5: A single ray-path reflective Schlieren optics. Uneven media present in the test area distort the light from the LED. Some rays, such as the red ray, will not reach the camera. High refraction makes the corresponding pixels darker than the normal area.

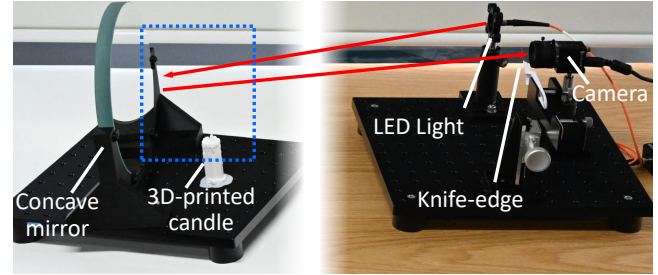


Figure 6: Our Schlieren setup for the experiments. The red ray is a schematic path of the light source. If inhomogeneous media sits on the path practically in the test area depicted by blue dotted square distort the light from the LED and block by the knife-edge, which becomes darker in Schlieren images. The distance between the concave mirror and the camera is 2 times the focal length of the mirror, which is 1.5m.

The choice of light source is another design option. A laser source can generate a short pulse. Thus, laser-based Schlieren is suitable for analyzing fast airflow like a jet, while it suffers from speckle noise as shown in Fig. 7. We thus employed LED-based Schlieren given that we uses PIV in the later step.

4.3 Software and Calibration

Given the capture setup, we analyze the captured Schlieren images in video rate. We then render the airflow for visualization and simulate physical effects in Unity3D.

For software we used a standard laptop (Intel i7-8700T CPU @ 2.40GHz and 16.0 GB RAM) with OpenCV [5] and OpenPIV [22] in python 3.9 for image analysis. For rendering and interactions, we adopt Unity3D as the game engine.

To balance the complexity of handling of the estimated airflow and the computation cost, we implemented an angle extract algorithm under the assumption that available airflows have the primary flow direction.

We first explain our PIV procedure to estimate the velocity distribution of the airflow using OpenPIV. We get bright dots images in the Schlieren images (2064 x 1544 pixels) by OpenCV required for PIV [22]. Then we set the size of split windows for PIV analysis as 24 pixels. After that, we compute the pixel to the distance of the real world by using the length of the concave mirror and the time between frames.

To estimate the density of the airflow for pressure estimation, we applied the scaling Schlieren method [23]. This method is based on the behavior of the intensity change of static air at the knife edge, i.e. the focus point of the light rays, when this occlusive edge is slightly shifted along the vertical plane of optical axis. Following the theory we calibrate the system offline and obtained the calibration curve. This calibration is required only once for each system.

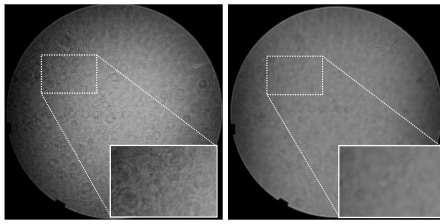


Figure 7: Comparison of two light sources: laser type on the left and LED type on the right. The Schlieren image with a laser light source degrades detection accuracy due to the speckle noise inherent in lasers, whereas the LED light source captures the background more clearly.

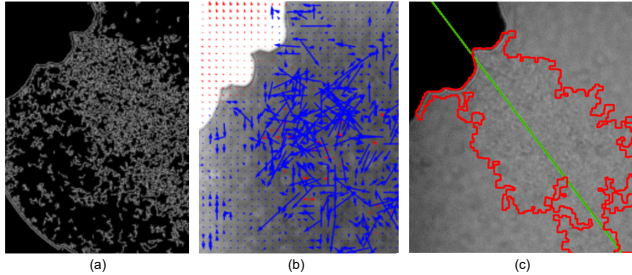


Figure 8: The preprocessed image and the analyzed image. (a) The preprocessed Schlieren image. (b) The PIV analyzed images that blue vectors are standing for the particle fluids' velocity of the airflow. The longer these blue vectors are, the fast they are. (c) The result of our angle extraction algorithm. This algorithm finds the outline of the airflow (red line) and extracts the outcoming angle (green line) by linear regression.

We collected 20 positions to complete the calibration curve as Fig. 9. Theoretically, the curve follows a linear model and its gradient value allows to compute the density from a new Schlieren image. We thus fit a line to this data set and got the gradient value of about 2.66. Collecting more data improves the density accuracy. For example, the related work collected 100 points. Yet, in our scenario with dynamic airflow, unlike their convective flow estimation scenario, our calibration with 20 points was sufficient because we only needed to get the gradient change of density for detecting the density differences of airflow.

When calculating the airflow, we use the image taken at a fixed knife-edge as the reference image at a certain position. Then we subtract the captured image with airflow from the reference image. The calibration curve is then used to convert the knife-edge distance at calibration from the gray intensity of each pixel in the subtracted image. Finally, we calculate the airflow density from the knife-edge distance, following [23].

This procedure also gives the density and the temperature. Now, we will use the density and the velocity from PIV analysis to simulate the airflow's physical effects in the AR environment.

The angle extraction is based on contour detection. To make the airflow complete as a connected contour and clean the noise as much as possible, we dilate and erode the images in proper iteration numbers. Once we find that there is airflow in the images, we will detect the longest contour in the images and calculate the angle using linear regression. In addition to the angle, we also compute the average density and pressure of the airflow within the mask region. Finally, we send this spatially-averaged airflow's angle, velocity, and pressure to Unity3D to visualize the particles.

Fig. 8 (a) is the preprocessed data, which we use in velocity

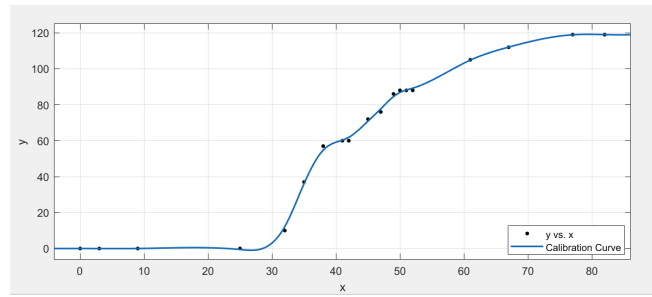


Figure 9: Calibration curve used for the density estimation. The y axis is the gray intensity and x is the knife position shift observed in the image space [pixel]. Measuring the gradient of the intensity change allows to compute the density via the scaling Schlieren method [23].

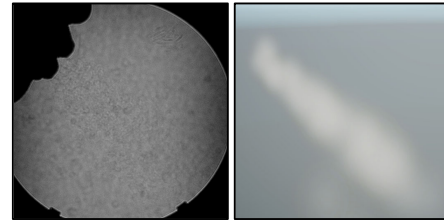


Figure 10: An example of visualized airflow from a real Schlieren image. Our pipeline extracts flow information from a Schlieren image and convert them into a particle object in Unity3D.

analysis and the analyzed result. We draw vectors of calculated results in the original image as in Fig. 8 (b). The longer blue vectors are, the faster the velocity they represent. Also, there are some red vectors that are filtered because the weight of cross-correlation is lower than the threshold we set before. The threshold depends on the sensor's sensitivity and light brightness and thus needs heuristic tuning. Moreover, we can set another threshold to filter the noise when a pixel's flow speed is too low. Fig. 8 (c) shows the result of our angle extraction algorithm. The angle is reasonable enough for interaction mostly. But we also observed when the airflow source is moving too fast, it makes the primary direction result unstable.

Game Engine We use the particle system in Unity3D to simulate the airflow in the real world. We also need to adjust the parameter of a particle object for visualization. To make a natural airflow in Unity3D, one needs to tune the particle system settings property. We heuristically optimized the settings while rendering basic texture of our particle objects. Fig. 10 is the final effect of our particle object.

5 EXPERIMENT AND DISCUSSION

To demonstrate our concept, we first verify our Schlieren implementation customized for AR scenarios in Sec. 5.1, and then explore applicability to AR scenarios in Sec. 5.2. We also conduct interaction analysis in Sec. 5.3 based on AR HCI principles [10] to provide the insight to help applying our concept in potential AR applications.

5.1 Schlieren System Evaluation

To make the Schlieren modality applicable to AR scenarios, we developed the custom estimation and visualization pipeline to be capable of real time computation while providing reasonable pressure and direction information of airflows. We thus evaluated and show that our pipeline can estimate these values correctly.

Initially, we experimented with the temperature and pressure of the airflow that a proof-of-concept system can measure. We used a

Table 1: The air volume [m^3/h] records of the heat gun’s document.

Volume Level	Air Volume	Percentage (%)
1	0.15	60
3	0.2	80
5	0.25	100

Table 2: The calculated pressure [N/m^2] and its percentages at 100°C.

Volume Level	Pressure	Percentage (%)
1	11.9556	91
3	12.693	96
5	13.13	100

heat gun (HAKKO FV-310) that can blow wind at 5 different levels. We chose three linear levels (0.15, 0.20, 0.25 m^3/min) of them compared with the computational result of our algorithm. From the documents of the heat gun, we recorded the air volume of levels 1, 3, and 5, then set them as percentages just like Table 1. Table 1 shows that the air volume of the heat gun varies linearly with the volume level of 5 as 100%. Thus, we measured airflow at temperatures of 100°C and 410°C, and examined the measured airflow changes linearly with respect to the volume level.

We collect 50 to 100 data of velocity and density at different levels. We related the air volume to the dynamic pressure of the airflow, so we compared the document’s record with the calculated pressure of our algorithm.

Table 2 and table 3 show the results. The results show that, the estimated values show linear tendencies with respect to the volume level as expected. Qualitatively speaking, we can see that the measured pressure increases with airflow at higher temperatures as expected.

5.2 Demonstrations

In addition to the pressure and temperature measurement, as we already showed in Fig. 10, our system also provides the spatial information of the airflow, i.e. directions, which is crucial for AR/VR uses. In this section, we demonstrate our proof-of-concept system via toy VR and AR applications.

5.2.1 VR Demo: Virtual Balls

We first showcase our spatial airflow measurement pipeline under a VR environment using the pressure and direction values of real airflows. Please also refer to our demo video to see the actual simulation.

For real airflows, we used the heat gun from the previous section with 410°C and air volume level 1, giving $15.8N/m^2$. For each trial, we set a virtual object at different masses and let them interact with real airflow measurements. The airflow was set at the origin. Given the heat gun’s pressure distributions, we placed a virtual rigid ball with a radius of 0.5 units in the space and a mass of 0.1, 1, 5, and 10kg each in four trials in the virtual space. In the rigid body property, we set drag to 0.5 and angular drag to 0.05. Since the spatial airflow is locked to the origin, the balls stop at some distances far enough from the airflows.

Table 4 shows the resulting distance when the ball stops after being moved by each airflow measurement. Even with the same airflow, we can add the quantitative value as a force to move the balls at different masses, and the moving distances vary according to the masses just like in the real world.

5.2.2 AR Demo: Virtual Candle Fire on Real Candle

We demonstrate another AR scenario: blowing out the fire of a virtual candle with a real breath, which is essentially not possible

Table 3: The calculated pressure [N/m^2] and its percentages at 410°C.

Volume Level	Pressure	Percentage (%)
1	15.8356	83
3	17.1102	90
5	18.9588	100

Table 4: Distance records with different ball’s masses.

Ball’s mass	0.1	1	5	10
Distance	8.3757	5.4594	3.1674	2.1964

via conventional AR interaction modality. Please also refer to our demo video to see the animated results.

Unlike the VR demo above, this concept demonstration shows that spatial airflow can interact with virtual objects in a real AR environment. We implemented a video see-through setup using Unity3D and an additional RGB camera. A color scene was captured by another RGB camera (A Google Pixel 6 Pro camera, no zoom) placed next to the one for the Schlieren imaging. We manually aligned the offset between the Schlieren airflow’s coordinates and the candle in the scene view.

As a real object, we printed a candle base model (a 3D model we took from a public repository [9]). And then we rendered a virtual fire using a renderer asset [36] and set interactive in Unity 3D.

In the demo, when one blows weakly to the virtual candle, the flame will just shake because the power is not enough. However, if users blow strongly, the flame will be extinguished.

In our game engine implementation, we set the durability of our virtual candle, and the pressure value from the airflow will reduce the durability. The size of the flame will become smaller and smaller if the durability decreases. However, if the durability does not become zero, then it will recover slowly and the flame will come back again. Thus, a person should blow strongly enough so that they can kill the flame and make it never recover. Note that the system theoretically can utilize airflow types other than breathing such as a hairdryer, a fan, and so on. In our test, we tried normal breath, weak breath, and a heat gun at level 3. When the pressure was strong enough, the virtual fire was extinguished. Otherwise, the fire will shake like Fig. 11.

Note that, since our system can also calculate the spatial information from Schlieren images, we simulated not only the pressure but also the angle of airflow. The fire will interact with airflow according to its pressure and direction. The fade process of the fire will be different if the pressure value and direction change, as shown in Fig. 12.

5.3 Discussion: Interaction Analysis

We provide a preliminary analysis of our concept’s usability based on AR HCI principles by Andreas et al. [10]. Andreas et al. claim important design principles and discuss how they relate to AR system design, intending to provide good examples of applying HCI design principles in an AR setting. In addition, these guidelines for developing AR interfaces give general hints on how to avoid certain usability problems, which is a moderately objective and comprehensive way to analyze a new AR and VR interaction method.

Our system quantitatively analyses the airflow interaction, which means there is no threshold bothering users, and every interaction works without thinking of a specific key. Interactions in the AR system should be easy to use, have no redundant operation, and have a low learning curve. Existing blowing interactions focus on classifying the types of breathing actions and mapping them as a specific key in the virtual environment, such as a controller’s O, X button. Users have to think twice when interacting in this kind of system; for example, they have to learn which button maps

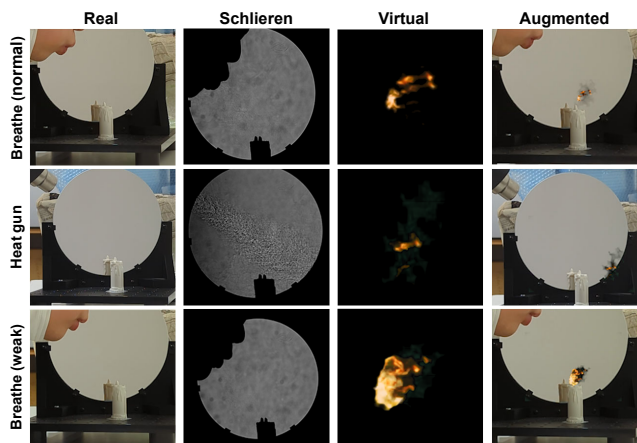


Figure 11: Our AR application demonstrations. Our system can detect different types of airflow quantitatively, such as breathing actions and a heat gun. We used the airflow’s spatially averaged velocity, density, and pressure.

which kind of breathing actions. Besides, these systems do the classification using algorithms with a threshold, which also adds the cost of learning and use to users. Even though the threshold can be proper enough, users will still consciously blow stronger than needed if they know there is a threshold. On the contrary, in our system, input is analog and relates to direct airflow behaviors. Therefore, it provides a low learning barrier for the users and generally increases the focus and immersion.

Objectively, we think existing blowing interactions and our method act in different roles. As mentioned before, our method can quantitatively detect the airflow that other works can not. If we adapt our method to a game like a tennis game, our method can help users, for example, charge the power bar instantly and determine how hard the character hits. However, other works cannot because they map breathing actions to a specific key. For example, users have to keep blowing. This holding action cannot be defined as a redundant operation because it also gives users a feeling that they are controlling the progress of the power bar. However, our method’s quantitative analysis can also offer users a kind of fun to weigh their power in the gameplay.

Interactions should also be flexible because of different user preferences and abilities. Our method can detect any airflow types captured by the camera by adopting Schlieren photography. This flexibility is another advantage over other existing blowing interactions.

6 LIMITATION AND FUTURE WORK

We discuss the limitations of the current prototype and possible future research directions.

6.1 Form-Factor

While our current benchtop system can detect and interact with airflow over a fixed space, it is attractive research question whether we can reduce the form factor applicable to small AR devices such as, ideally, HMDs. We currently adopt the reflective Schlieren system, which takes up space due to the concave mirror’s focal length and size. Background-oriented Schlieren optics shown in Fig. 2 (b) is another option. In this method, PIV is achieved by using a single camera to detect background fluctuations. Currently, a specific background pattern image must be needed in background-oriented Schlieren optical systems, and extending this to arbitrary backgrounds is one future research direction.

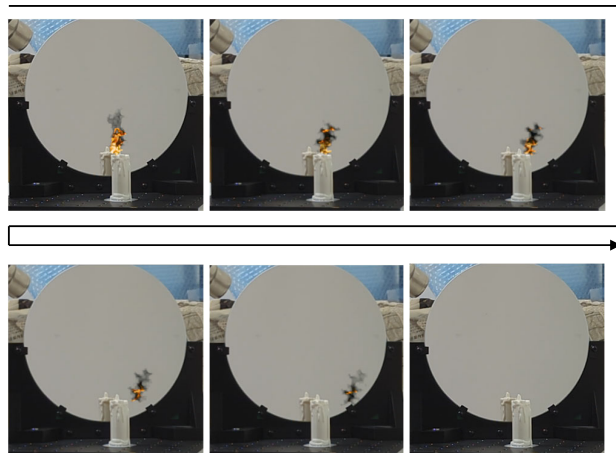


Figure 12: Time-series image of AR airflow interaction. A virtual fire is extinguished by a virtual airflow generated based on the measured airflow.

On the other hand, embedding Schlieren system can safely be justified when combined with spatial AR (SAR). Because SAR relies on projectors and cameras installed in the environment, it perfectly aligns with the natural idea of installing Schlieren systems in the environment. Allowing installation of the Schlieren system for AR allows the system to employ bigger several systems and bigger optics leading to a larger capture area and sensing accuracy.

6.2 Extending Detectable Properties

While we applied only the angle, velocity, and pressure of airflow to the interaction, theoretically, Schlieren imaging can quantitatively detect other airflow properties, such as temperature and density, in a spatial manner, allowing interaction with more complex 3D shapes. Also, Schlieren optics can get 3D spatial information by adopting two or more cameras. If enough computation resource is available, even reconstructing every particle fluid inside the airflow would be possible, as explored in the aerodynamics field. Once these extensions are realized, we consider this system can visualize the invisible airflow from a first-person perspective, which benefits wide industrial or daily-life applications, such as ventilation state of a room or the flow, or airflow simulation for industrial products such as an airplane.

6.3 Sensitivity Limitation

Overall, we think our airflow interaction system can detect the difference in pressure value sensitively, but the accuracy of sensitivity still needs to be improved. We supposed there will be two methods to improve: (1) design an algorithm specifically for Schlieren velocimetry, and (2) optimize our contour detection algorithm.

We can see from Table 5 that the sensitivity of velocimetry is rather low since the difference is minute. The main reason for this might be the PIV algorithm is mainly for seeded particles velocimetry. Even though we can adapt it for Schlieren measurements, there are still some limitations. For example, the PIV algorithm requires more than 5 bright dots in every small window for cross-correlation analysis, however, particles of Schlieren images sometimes close together and combine as a big bunch of airflow, which makes it hard to do image matching. Besides, it is also possible to change a higher frame rate and resolution camera to improve the particle motion capturing at the cost of computation requirement.

Also, this accuracy can be improved when the temperature of the airflow is higher. We supposed it is because the sensitivity of

Table 5: Velocity [px/s] records of two experiments.

Volume Level	Velocity(100°C)	Velocity(410°C)
1	0.722	0.7587
3	0.7372	0.7721
5	0.7429	0.7837

Table 6: Density [kg/m³] records of two experiments.

Volume Level	Density(100°C)	Density(410°C)
1	22.9349	27.5103
3	23.3559	28.7017
5	23.7906	30.8682

density detection becomes better when the temperature increase. We can see from Table 6 that at a higher temperature condition, the differences in density become more obvious, and the contours of airflow will become clearer because the gray intensity of the airflow will also increase. This makes the airflow become more separate from the background and easier to be detected, which also makes the background not going to mix in the contours and affect the result. For AR uses, however, using hotter airflow might not be ideal. An alternative option could be using a different media. For example, a dust spray that uses a gas of a larger refractive index also improves the density difference thus the sensitivity.

6.4 Interaction Limitation

In the real world, it takes time for particles to get to the objects they hit. However, since we simulated the airflow using the particle system in Unity3D and cannot fully control every single particle in this particle rendering pipeline, sometimes the interaction in Unity3D will happen before the airflow hits the objects. Thus, all values will be updated as soon as the airflow shows in the Schlieren images. For example, the airflow hit the ball earlier than real physics. This causes some issues that when we simulate the physical effect, the values we use are calculated from the first few frames.

To solve this issue, we think it is necessary to adopt a GPU particle rendering system, which can help us control every single particle in the airflow and simulate interaction more comprehensively. If we render the airflow using a GPU rendering pipeline, the whole airflow will not suddenly show up in Unity3D but will blow out step by step just like in the real world. Also, the particle will not hit the virtual object immediately.

Currently, our method seeks a single kinetic particle that is representative of the observed airflow. By leveraging the knowledge of the computer graphics community, as a future research direction, we could generate multiple kinetic particles from our computed airflow vector fields for AR interaction with more complex real-world scenes [6, 13].

However, even if we can measure the spatial vector field of airflow, there is an important point to consider for AR applications—the real airflow itself does not physically interact with virtual objects. Essentially, when airflow interferes with a real object, the airflow itself should change its distribution. However, since no such change occurs between the airflow and the virtual object, the naturalness of the interaction may be compromised. This is analogous to the difficulty of creating an AR display system that physically touches an AR object. Therefore, it is expected that some new innovations, for example, correcting the measured vector field over time according to the interference state between the AR object and the airflow vector field, will be a future prospect for this research field.

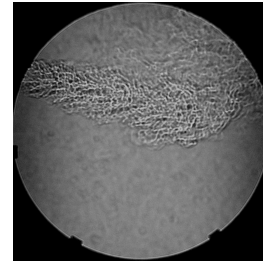


Figure 13: Robustness limitation in the current system (Sec. 6.6). When airflow source moves too fast, the distortion of airflow in Schlieren images will make the algorithm become incapable to calculate velocity, because the difference between two images is too big.

6.5 Computation Time

Considering responsiveness, every calculation of our system costs 0.2s on average. The rendering of the airflow is fluent in Unity3D, but it still takes time. The system cannot run totally in real-time, however, the execution time can be decreased by optimizing the algorithm or adopting multi-processing calculations, and the performance of this system will be more powerful with the development of computation ability.

6.6 Robustness of the System

The current Schlieren measurement system based on the PIV algorithm cannot detect the airflow when airflow source moves too fast (Fig. 13). The PIV algorithm needs two frames and builds a relation of two images for calculations. If the difference between the two images is too big, it is difficult to build a relationship and come out of a result. This issue can be solved if we adopt a higher frame rate camera. We can also consider adopting background-oriented Schlieren as our sensor for higher integration and a bigger detecting space.

7 CONCLUSION

In this work, we proposed a spatial airflow interaction method for AR using Schlieren photography, which overcomes limitations in existing blowing interactions. Our method can achieve detecting airflow's properties quantitatively and real-time interaction at a latency of 0.2s on average in the AR/VR environment. Unlike existing blowing interactions, our system can detect human breathing actions and any other types of airflow. Also, the Schlieren sensor model we use has high potential and high integration for AR and VR devices even with lower cost.

We demonstrated our concept in a VR and an AR scenario. The result shows the sensitivity of this system is good enough for interactions, and it still can improve for better performance. We also verified the system could finish airflow interactions we hypothesized in the game scene, which existing blowing interactions cannot. We explored the use cases of our system and analyzed the pros and cons compared with existing blowing interactions. In the end, we pointed out the future development of our system and discussed the specific solution.

ACKNOWLEDGMENTS

This project was partially supported by JST FOREST Grant Number JPMJFR206E, JST PRESTO Grant Number JPMJPR17J2, and JSPS KAKENHI Grant Number JP17H04692, and JP20H05958, Japan.

REFERENCES

- [1] A. K. Agrawal, N. Butuk, S. R. Gollahalli, and D. W. Griffin. Three-dimensional rainbow schlieren tomography of a temperature field in gas flows. *Applied optics*, 37 3:479–85, 1998.

- [2] C. Alvarez Herrera, D. Moreno-Hernández, B. Barrientos-García, and J. A. Guerrero-Viramontes. Temperature measurement of air convection using a schlieren system. *Optics & Laser Technology*, 41, 04 2009. doi: 10.1063/1.2926987
- [3] R. Beermann, L. Quentin, A. Pösch, E. Reithmeier, and M. Kästner. Background oriented schlieren measurement of the refractive index field of air induced by a hot, cylindrical measurement object. *Applied Optics*, 56:4168–4179, 05 2017. doi: 10.1364/AO.56.004168
- [4] T. Birch. *The History of the R. Society of London for Improving of Natural Knowledge from Its First Rise. As a Suppl. to the Philosophical Transactions.-London, A. Millar 1756-1757*, vol. 3. A. Millar, 1757.
- [5] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.
- [6] R. Bridson. *Fluid simulation for computer graphics*. AK Peters/CRC Press, 2015.
- [7] Y. Chen, Y. Bian, W. Gai, and C. Yang. Adaptive blowing interaction method based on a siamese network. *IEEE Access*, 8:115486–115500, 2020. doi: 10.1109/ACCESS.2020.3004349
- [8] M. K. Delimayanti, B. Purnama, N. G. Nguyen, K. R. Mahmudah, M. Kubo, M. Kakikawa, Y. Yamada, and K. Satou. Clustering and classification of breathing activities by depth image from Kinect. In *BIOINFORMATICS*, pp. 264–269, 2019.
- [9] dreadminis. 28mm candle, 2020.
- [10] A. Dünser, R. Grasset, H. Seichter, and M. Billinghurst. Applying hci principles to ar systems design. 2007.
- [11] J. Feijó Filho, T. Valle, and W. Prata. Breath mobile: A software-based hands-free and voice-free breathing controlled mobile phone interface. In *Proceedings of the 14th International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '12, p. 217–218. Association for Computing Machinery, New York, NY, USA, 2012. doi: 10.1145/2384916.2384961
- [12] I. Grant. Particle image velocimetry: a review. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 211(1):55–76, 1997.
- [13] J. Gregson, I. Ihrke, N. Thuerey, and W. Heidrich. From capture to simulation: connecting forward and inverse problems in fluids. *ACM Transactions on Graphics (TOG)*, 33(4):1–11, 2014.
- [14] M. J. Hargather and G. S. Settles. A comparison of three quantitative schlieren techniques. *Optics and Lasers in Engineering*, 50(1):8–17, 2012.
- [15] C. Jácome, J. Ravn, E. Holsbø, J. C. Aviles-Solis, H. Melbye, and L. Ailo Bongo. Convolutional neural network for breathing phase detection in lung sounds. *Sensors*, 19(8):1798, 2019.
- [16] D. Jain, M. Sra, J. Guo, R. Marques, R. Wu, J. Chiu, and C. Schmandt. Immersive terrestrial scuba diving using virtual reality. pp. 1563–1569, 05 2016. doi: 10.1145/2851581.2892503
- [17] P. S. Kolhe and A. K. Agrawal. Density measurements in a supersonic microjet using miniature rainbow schlieren deflectometry. *AIAA journal*, 47(4):830–838, 2009.
- [18] K. Lasinger, C. Vogel, T. Pock, and K. Schindler. 3d fluid flow estimation with integrated particle reconstruction. *Lecture Notes in Computer Science*, 2019.
- [19] J. N. Latta and D. J. Oberg. A conceptual virtual reality model. *IEEE Computer Graphics and Applications*, 14(1):23–29, 1994.
- [20] Z. Li, Y. Ji, J. Yu, and J. Ye. 3d fluid flow reconstruction using compact light field piv. In *ECCV*, 2020.
- [21] Z. Li, J. Ye, Y. Ji, H. Sheng, and J. Yu. Piv-based 3d fluid flow reconstruction using light field camera. *ArXiv*, abs/1904.06841, 2019.
- [22] A. Liberzon, T. Käufer, A. Bauer, P. Vennemann, and E. Zimmer. Openpiv/openpiv-python: Openpiv-python v0.23.4, Jan. 2021. doi: 10.5281/zenodo.4409178
- [23] A. Martinez Gonzalez, D. Moreno-Hernández, and J. A. Guerrero-Viramontes. Measurement of temperature and velocity fields in a convective fluid flow in air using schlieren images. *Applied optics*, 52:5562–9, 08 2013. doi: 10.1364/AO.52.005562
- [24] W. Merzkirch. *Flow visualization*. Elsevier, 2012.
- [25] K. Y. Nagulin, D. S. Akhmetshin, A. K. Gilmudinov, and R. A. Ibragimov. Three-dimensional modeling and schlieren visualization of pure ar plasma flow in inductively coupled plasma torches. *Journal of Analytical Atomic Spectrometry*, 30(2):360–367, 2015.
- [26] H. Ohno and K. Toya. Scalar potential reconstruction method of axisymmetric 3d refractive index fields with background-oriented schlieren. *Optics Express*, 27(5):5990–6002, 2019.
- [27] S. Patel and G. Abowd. Blui: Low-cost localized blowable user interfaces. pp. 217–220, 01 2007. doi: 10.1145/1294211.1294250
- [28] R. Patibanda, F. F. Mueller, M. Leskovsek, and J. Duckworth. Life tree: Understanding the design of breathing exercise games. CHI PLAY '17, p. 19–31. Association for Computing Machinery, New York, NY, USA, 2017. doi: 10.1145/3116595.3116621
- [29] J. Rienitz. Schlieren experiment 300 years ago. *Nature*, 254(5498):293–295, 1975.
- [30] E. Sawada, S. Ida, T. Awaji, K. Morishita, T. Aruga, R. Takeichi, T. Fujii, H. Kimura, T. Nakamura, M. Furukawa, et al. Byu-byu-view: a wind communication interface. In *ACM SIGGRAPH 2007 emerging technologies*, pp. 1–es. 2007.
- [31] G. Settles and M. Hargather. A review of recent developments in schlieren and shadowgraph techniques. *Measurement Science and Technology*, 28, 01 2017. doi: 10.1088/1361-6501/aa5748
- [32] M. Slater and M. Usoh. Body centred interaction in immersive virtual environments. *Artificial Life and Virtual Reality*, 06 1999.
- [33] T. Sonne and M. M. Jensen. Chillfish: A respiration game for children with ADHD. *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction*, 2016.
- [34] F. Sourgen, F. Leopold, and D. Klatt. Reconstruction of the density field using the colored background oriented schlieren technique (cbos). *Optics and Lasers in Engineering*, 50(1):29–38, 2012.
- [35] M. Sra, X. Xu, and P. Maes. BreathVR: Leveraging breathing as a directly controlled interface for virtual reality games. *2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2018.
- [36] H. Studio. 3d fire and explosions, 2020.
- [37] W. Thielicke and E. J. Stambhuis. Pivlab – towards user-friendly, affordable and accurate digital particle image velocimetry in matlab. *Journal of Open Research Software*, 2, 10 2014. doi: 10.5334/jors.bl
- [38] J. D. Tobin and M. J. Hargather. Quantitative schlieren measurement of explosively-driven shock wave density, temperature, and pressure profiles. *Propellants, Explosives, Pyrotechnics*, 41(6):1050–1059, 2016.
- [39] E. Traldi, M. Boselli, E. Simoncelli, A. Stancampiano, M. Gherardi, V. Colombo, and G. S. Settles. Schlieren imaging: a powerful tool for atmospheric plasma diagnostic. *EPJ Techniques and Instrumentation*, 5:1–23, 2018.
- [40] G. Wang. Designing smule's ocarina : The iphone's magic flute. 01 2009.
- [41] D. Zielasko, S. Freitag, D. Rausch, Y. Law, B. Weyers, and T. Kuhlen. Blowclick: A non-verbal vocal input metaphor for clicking. 08 2015.