A Framework for the Generation of Textures Representing Time-Dependent Changes in the Appearance of Dust Layers

by

Rebecca L. C. Santos

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2022

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

The perception of realism in computer generated images can be significantly enhanced by subtle visual cues. Among those, one can highlight the presence of dust on synthetic objects, which is often subject to temporal variations in real settings. Moreover, by realistically depicting the appearance of dust accumulated over time, one can also convey subtle information about the history of a scene. In this thesis, we present a framework for the generation of textures representing the accumulation of this ubiquitous material over time in indoor settings. It employs a physically-inspired approach to portray the effects of different levels of accumulated dust roughness on the appearance of substrate surfaces, and to modulate these effects according to the different illumination and viewing geometries. The development of its core algorithms was guided by empirical insights and data obtained from observational experiments which are also described. To illustrate its applicability to the rendering of visually plausible depictions of time-dependent changes in dusty scenes, we provide sequences of images obtained considering distinct dust accumulation scenarios.

Acknowledgements

I am deeply grateful to my supervisor, Professor Baranoski, for his encouragement and guidance throughout the entirety of my program. I would like to thank Professor Alencar and Professor Daudjee for taking the time to read my thesis and offer meaningful feedback. I would also like to offer my thanks to the other members of the Natural Phenomena Simulation Group - Petri Varsa, Spencer Van Leeuwen, and Mark Iwanchyshyn. Beginning with aiding in my decision to pursue a Master's, all the way to listening to my practice presentation, your support was incredibly valuable. Finally, I would like to thank my partner, Christopher, and my mother. Your confidence helped spur me to the finish line.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Despite significant advances in realistic image synthesis in recent years, notably involving the simulation of weathering and aging phenomena (*e.g.*, [5, 31, 35, 38, 44]), computer generated images are often characterized by a "clean look". Real settings, however, have a less pristine appearance. This can be perceived, for example, through visual cues resulting from time-dependent phenomena such as the accumulation of dust (Fig. 1.1), the focal point of this work. By accounting for the presence of this pervasive material and the temporal variations in its appearance, one can considerably improve the degree of realism of synthetic settings.

In order to effectively incorporate the presence of dust in synthetic settings, one should select an appropriate level of abstraction to represent its intrinsic complexity. For example, household dust can be composed of a wide range of materials, from textile fibres and hair strands to human skin fragments and mineral particles, just to name a few [48]. Moreover, the location of a household dust deposit can also affect its composition, notably in areas prone to variations in indoor environmental conditions (*e.g.*, ventilation changes [49]).

1
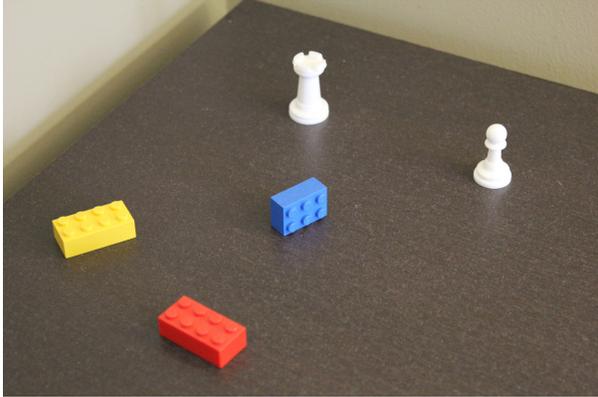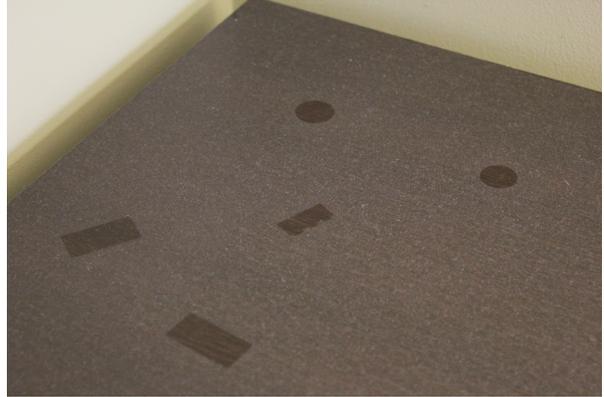
Figure 1.1: Photos of real settings depicting the accumulation of dust over a period of several months. (a) and (c): photos taken before the removal of objects from the settings. (b) and (d): photos taken after the removal of the objects.

Ideally, the selected level of abstraction should match the requirements of the rendering application in which it will be used.

A number of works addressing the rendering of dusty scenes have been presented in the literature (*e.g.*, [23, 26, 47]). To correctly simulate the appearance of dust using

physically-based approaches, one needs to correctly simulate complex light interactions with its distinct constituent materials. This often requires precise information about their spatial and size distribution patterns. Although such approaches may result in predictive images [22], they can potentially incur relatively high computational costs. Moreover, their effectiveness may be limited by the scarcity of measured optical and morphological data associated with the broad range of dust's constituent materials.

Physically-inspired approaches, on the other hand, are aimed at the generation of believable images. Although such images are not meant to be predictive, they can provide plausible representations of a given setting without one having to carry out potentially costly light transport simulations. Thus, although they may have a limited use in scientific applications requiring predictability, they are often adequate for applications with educational or entertainment purposes.

It is worth noting that physically-inspired (instead of physically-based) methods are extensively employed in different areas of computer graphics (*e.g.*, [4, 40, 45, 51]). These methods, albeit inspired by real physical observations, are developed from a relaxation of physically-based methodology in order to avoid complex formulations and, consequently, keep the representation of different materials plausible at a relatively low cost. For the same reasons, we also elected to employ a physically-inspired approach in our work.

In this thesis, whose main contributions have been also disseminated in an associated journal publication [43], we present a novel framework for the generation of textures to be employed in the depiction of temporal changes in the appearance of dusty objects placed in indoor settings. The design of its core algorithms, albeit building on methodology adopted by existing physically-inspired approaches, was driven by actual observations of dust accumulation over time. This strategy enabled us to strive for a high degree of plausibility while keeping the associated computational costs relatively low and the algorithms' incorporation

3

into existing rendering systems straightforward.

Accordingly, for the purposes of the proposed framework, we categorize dust's composition into two roughness levels, one associated with an opacity film (formed by fine dust particles) and the other with a granularity mask (containing coarse dust elements like particle clumps and hairy structures). These abstraction levels, which are based on methodical visual examinations of deposited dust, are represented by textures. During rendering, these textures are combined taking into account the dependence of dust appearance on viewing and illumination geometries. The time-dependent specification of textures is supported by data collected from observational experiments on dust accumulation, which have been carried out over extended periods of time. To the best of our knowledge, rendering algorithms that simultaneously account for different levels of dust roughness and their temporal variations have not been presented in the computer graphics literature to date.

The remainder of this thesis is organized as follows. In Chapter 2, we briefly present related works on the accumulation and rendering of dust. In Chapter 3, we highlight relevant aspects associated with the perceived appearance of accumulated dust. In Chapter 4, we outline the procedures employed to obtain supporting data on dust accumulation over time. In Chapter 5, we describe the proposed framework's texture generation algorithms. In Chapter 6, we illustrate its applicability to realistic image synthesis and concisely address possible extensions to its formulation. Finally, in Chapter 7, we conclude the thesis and point out directions for future research related to this topic.

Chapter 2

Related Work

In this chapter, we concisely review relevant publications that have specifically addressed the accumulation and rendering of dust. It is worth noting that there is a large body of work on topics that could be connected to the research presented in this thesis. Besides the simulation of weathering and aging phenomena mentioned earlier, these topics include the modeling and rendering of granular materials, such as snow and sand (*e.g.*, [7, 19, 32, 33, 37], as well as the simulation of dust dynamics (*e.g.*, [3, 10, 12, 13, 27]), among several others. A comprehensive review of the literature on these topics, however, would merit an entire publication devoted to it, which is beyond the scope of this work.

2.1 Dust Accumulation

Hsu and Wong [26] proposed an algorithm to simulate the accumulation of dust that employs the dot product of a target surface's normal vector and the normal vector of a user-defined dust source to determine the amount of deposited material. They also considered

external factors that may affect the amount such as scraping off by other objects. The dust accumulation patterns resulting from the application of their algorithm are stored as texture maps. Lu *et al.* [35] indicated that the algorithm proposed by Hsu and Wong [26] could be combined with other techniques, notably ambient occlusion[1], to obtain physically plausible context-aware textures depicting the presence of dust.

Afterwards, Chen *et al.* [15] proposed an algorithm for handling material lifespan changes leading to rusty and dusty appearances. They also made use of the target surfaces' normal vector, while incorporating additional weathering concepts such as user-defined material deterioration functions.

More recently, Guo and Pan [23] introduced an algorithm based on the application of shadow mapping and ambient occlusion techniques to determine the amount of dust at each point on a target surface. We note that the dust accumulation patterns resulting from the application of these algorithms are relatively uniform, *i.e.*, certain roughness aspects, such as the presence of dust particle clumps and fibrous (hairy) structures, are not normally explicitly incorporated into the simulations.

2.2   Dust Rendering

Once the dust has been placed within a scene (*e.g.*, deposited on the surface of an object), it must be rendered. Most of the algorithms employed in the rendering of accumulated dust can be classified as *ad doc*, *i.e.*, they are created and used for that particular purpose. These algorithms usually rely on the use of texture maps and noise functions. Accordingly, individual dust particles and particle clumps are seldom explicitly accounted for. Although

---

[1]A method used to determine the visibility of objects within a rasterized scene, adding depth with ambient lighting [39].

these algorithms are not intended to follow a physically-based approach, they can produce believable results without requiring extensively time-consuming computations.

For instance, in the work by Hsu and Wong [26], the user must define materials at two extremes, namely covered by dust and devoid of dust. From there, the algorithm linearly interpolates between these two extremes according to the amount of dust calculated previously. Hsu and Wong [26] also suggested that the appearance of the rendered dusty materials could be improved by adding Perlin noise[2].

Similarly, Chen *et al.* [15] proposed an algorithm that relies on a user-defined custom deterioration function (*e.g.*, involving changes in the diffuse and specular components of a material's reflective behaviour). This function is then incorporated into the user's shading method of choice. Also, Ashraf and Chong [3] used a 3D fractal surface shader[3] to imitate a thin layer of dust.

Adachi *et al.* [1] modeled thick layers of dust through the combined use of parametric shell textures and Perlin noise [42]. In contrast to the previously cited works, their algorithm is focused on the fibrous component of those layers. It is worth mentioning that, despite the fact that many of these *ad hoc* algorithms rely on the availability of suitable texture maps, methods to obtain physically plausible dust textures, such as those proposed by Hoock [25] and Ofek *et al.* [41], are still scarce in the literature.

Although physically-based approaches specifically tailored to the rendering of dusty materials have not been extensively explored by the computer graphics community, there are noteworthy initiatives in this area. For example, building up on the algorithms pre-

---

[2]A type of gradient noise proposed by Perlin [42].

[3]Shaders can be seen as modular programs that encapsulate illumination and shading calculations employed by a rendering system [20]. This modularization approach allows for efficient execution of these calculations on graphics processing units (GPUs).

sented by Blinn [8], Sun *et al.* [47] proposed a framework for the acquisition and rendering of time-varying changes in the reflective behaviour of different materials, including dusty surfaces. Guo and Pan [23], on the other hand employed a model based on the Kubelka-Munk theory [29, 34] to render settled dust. Again, we note that these physically-based approaches also resulted in relatively uniform representations of a layer of accumulated dust.

Chapter 3

Empirical Background

We remark that the algorithms proposed in this thesis are based on insights obtained from careful visual assessments of real dusty settings. In this chapter, we present observed qualitative characteristics that provided the basis for the selection of the distinct levels of abstraction used to characterize a dust layer in this work, namely the opacity film and the granularity mask. Recall that we employ the former to represent fine dust particles, and the latter to represent coarse dust elements such as particle clumps and hairy structures.

Although our observations involved a number of distinct settings, for conciseness and without loss of generality, we selected a representative case to be used as an observation reference in this chapter. It consists of a binder clip on a flat surface as depicted in the photos presented in Fig. 3.1. More specifically, we use this setting to verify how the perception of the different levels of dust roughness is affected by variations on the viewing and illumination geometries, which have as their main parameters the viewing angle (denoted by θ_v) and the light incidence angle (denoted by θ_i), respectively. For consistency, both angles are measured with respect to the flat surface's normal vector.
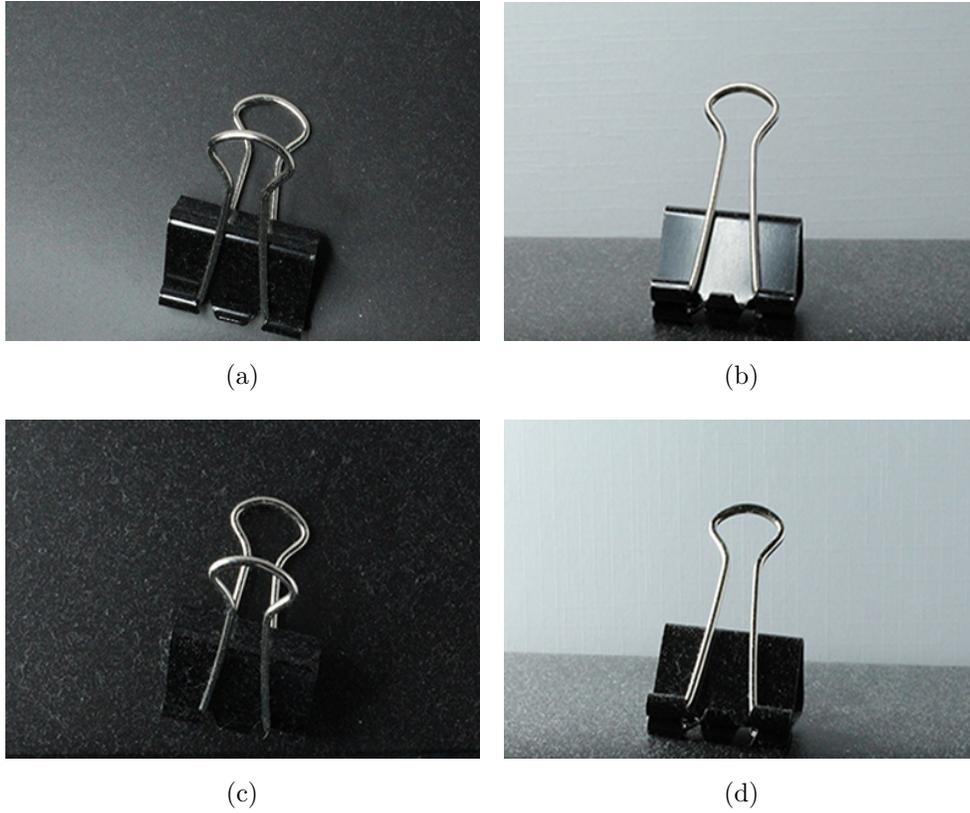
Figure 3.1: Photos of a dusty scene depicting the visual effects of variations in the viewing and illumination geometries. The photos were taken considering distinct viewing (θ_v) and light incidence (θ_i) angles measured with respect to flat surface's normal vector. (a) $\theta_v = 0°$ and $\theta_i = 0°$. (b) $\theta_v = 80°$ and $\theta_i = 0°$. (c) $\theta_v = 0°$ and $\theta_i = 80°$. (d) $\theta_v = 80°$ and $\theta_i = 80°$.

As shown in the photos presented in Figs. 3.1(a) and (c), large scale features, such as particle clumps and hairy structures, are clearly noticeable upon a top view ($\theta_v = 0°$) inspection of the scene, while fine dust particles are less prominent. As the viewing angle is increased to enable a grazing view ($\theta_v = 80°$) inspection of the scene, the film of fine dust particles becomes more noticeable as shown in Figs. 3.1(b) and (d), while variations

in the perceived appearance of large scale features are negligible for practical purposes.

Recall that the photos presented in Fig. 3.1 were obtained considering two light incidence angles, $\theta_i = 0°$ and $\theta_i = 80°$. By considering a large light incidence angle, which corresponds to the placement of the light source in a grazing position, one can make both levels of dust roughness more noticeable. For instance, one can observe more clearly the relatively nonuniform shape of the particle clumps and hairy structures in Fig. 3.1(c) (in comparison with Fig. 3.1(a)), as well as the overall presence and light reflective properties of the film formed by fine dust particles in Fig. 3.1(d) (in comparison with Fig. 3.1(b)).

Clearly, the appearance of a dusty surface depends on all of the different levels of dust roughness. However, our observations indicate that, for analysis and rendering purposes, a dust layer can be virtually decomposed into the two abstract representations employed in this work, namely the opacity film and the granularity mask, as illustrated by the diagram presented in Fig. 3.2. Moreover, these observations also indicate that the contributions of these two representations need to be properly modulated, by taking into account the viewing and illumination geometries employed in a given scene, in order to realistically depict the appearance of dusty surfaces. These aspects are accounted for in the formulation of the algorithms proposed in this work (Chapter 5).

substrate surface   dust layer   dusty surface

opacity film   granularity mask
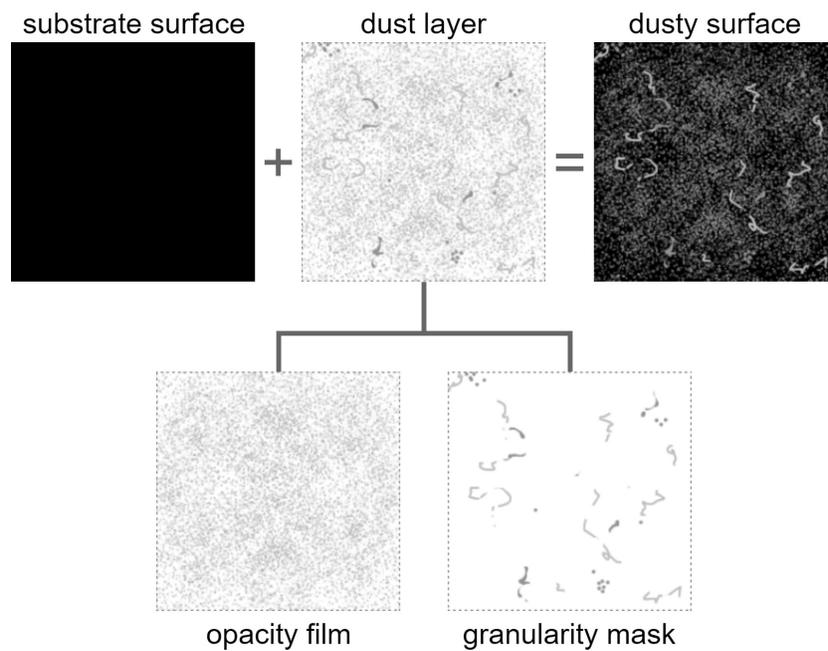
Figure 3.2: Diagram illustrating the contributions of fine and coarse dust components, respectively represented by an opacity film and a granularity mask, to the appearance of dusty surfaces.

Chapter 4

Dust Accumulation Experiments

Besides the aspects outlined in the previous chapter, we also carried out a number of observational experiments to obtain additional supporting information for the design of the proposed texture generation framework (Chapter 5). In the remainder of this chapter, we briefly describe the procedures used to gather and analyze these datasets.

Again, for conciseness and without loss of generality, we present here two representative cases of our observational experiments. They involved the daily photographic record of dust deposited in two distinct real-life settings (Fig. 4.1). Although we have initially considered a temporal resolution in terms of hours instead of days, we found that the latter allows us to obtain more meaningful insights since hourly variations on dust accumulation are normally negligible upon visual inspection.

The analysis of the resulting photos, which involved the quantification of visually-relevant dust components (fine particles and coarse elements) present in each setting on each day, provided us with a baseline for the design of the algorithms employed by the proposed framework. More specifically, we were able to obtain approximated time-dependent

<center>(a)   (b)   (c)</center>

<center>(d)   (e)   (f)</center>
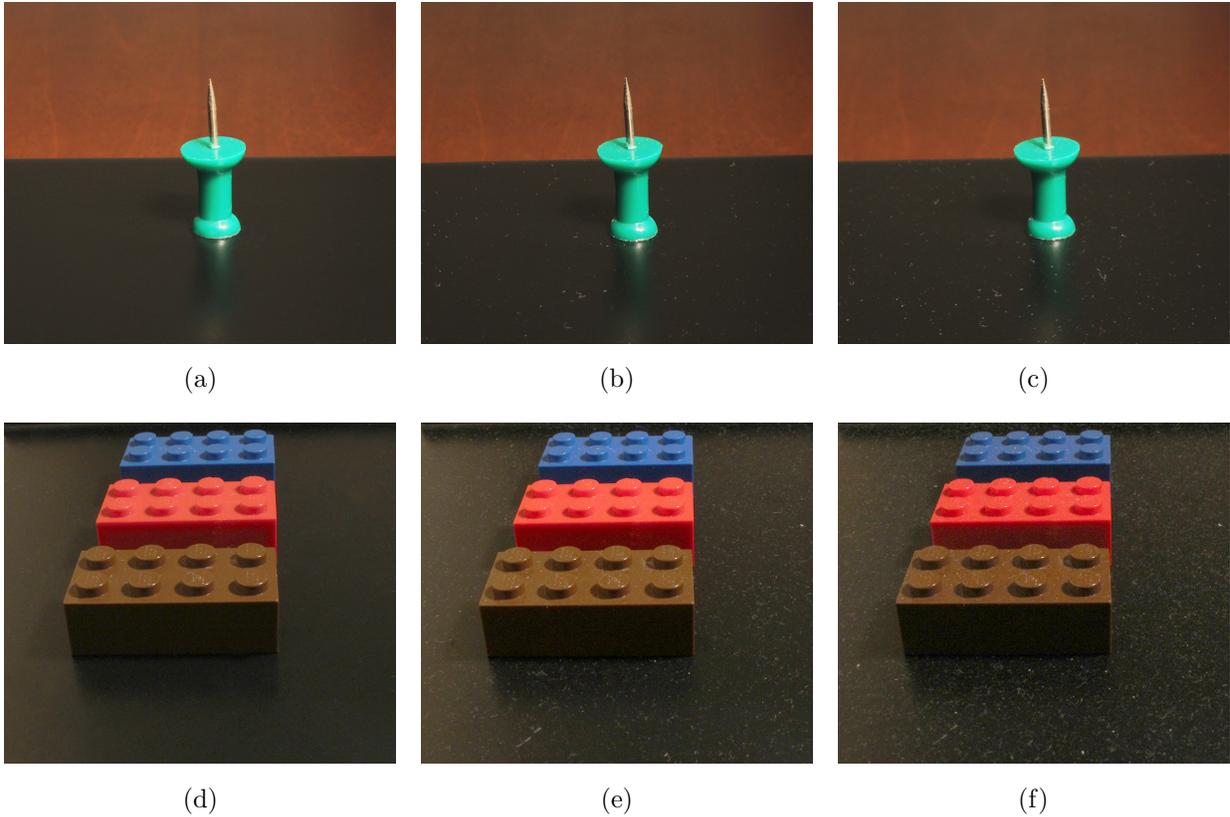
Figure 4.1: Photos depicting two real settings employed in our observational experiments on dust accumulation over time. The thumbtack scene (top row) was set in a room with a low level of human presence and activity, while the lego scene (bottom row) was set in a room with high level of human presence and activity. (a) and (d): initial state. (b) and (e): after 10 days of dust accumulation. (c) and (f): after 20 days of dust accumulation.

descriptions of the dust accumulation processes. These descriptions, in turn, were employed to guide the generation of texture representing accumulated dust in synthetic scenes at different time instances (Chapter 5).

The selected settings, which are shown in Fig. 4.1, included distinct objects placed in

<center>14</center>

environments with distinct levels of human presence and activity so that we could increase our scope of observations. The central positioning of the objects (thumbtack and legos) on a flat dark surface enabled consistent focal length between photos taken at different time instances. This surface was selected in order to enable a high degree of fidelity in the visual detection of dust elements. The thumbtack scene (Figs. 4.1(a), (b) and (c)) was set in a room with a low level human presence and activity, while the lego scene (Figs. 4.1(d), (e) and (f)) was set in a room with high level of human presence and activity.

After taking the photos, fine particles and coarse dust elements on the dark flat surface were marked and counted using standard image processing tools (Appendix A). The number of coarse dust elements, such as relatively large clumps and hairy structures, were compounded to inform the generation of the granularity masks, whereas the number of fine particles was employed to inform the generation of the opacity films used by the proposed framework (Chapter 5). We remark that we chose to adopt a physically-inspired approach in the design of the proposed algorithms. Accordingly, our observational experiments and the analysis of their outcomes were aimed at obtaining approximate estimates of the relative number of dust elements, rather than precise readings.

To facilitate comparisons between the outcomes of the observational experiments associated with the two photographed settings, we employed a simple normalization procedure. It consisted in dividing the total numbers of marked dust elements for each day by the respective areas of the dark flat surface considered during the marking operation carried out for each setting. This, in turn, resulted in total dust element numbers normalized per in^2 (or $6.4516\ cm^2$) of the dark flat surface. The obtained values along with their respective approximated trends over time are shown in the graphs presented in Fig. 4.2. We note that the approximated trends, which are presented for illustrative purposes, were obtained using the least squares method [9, 24].

15
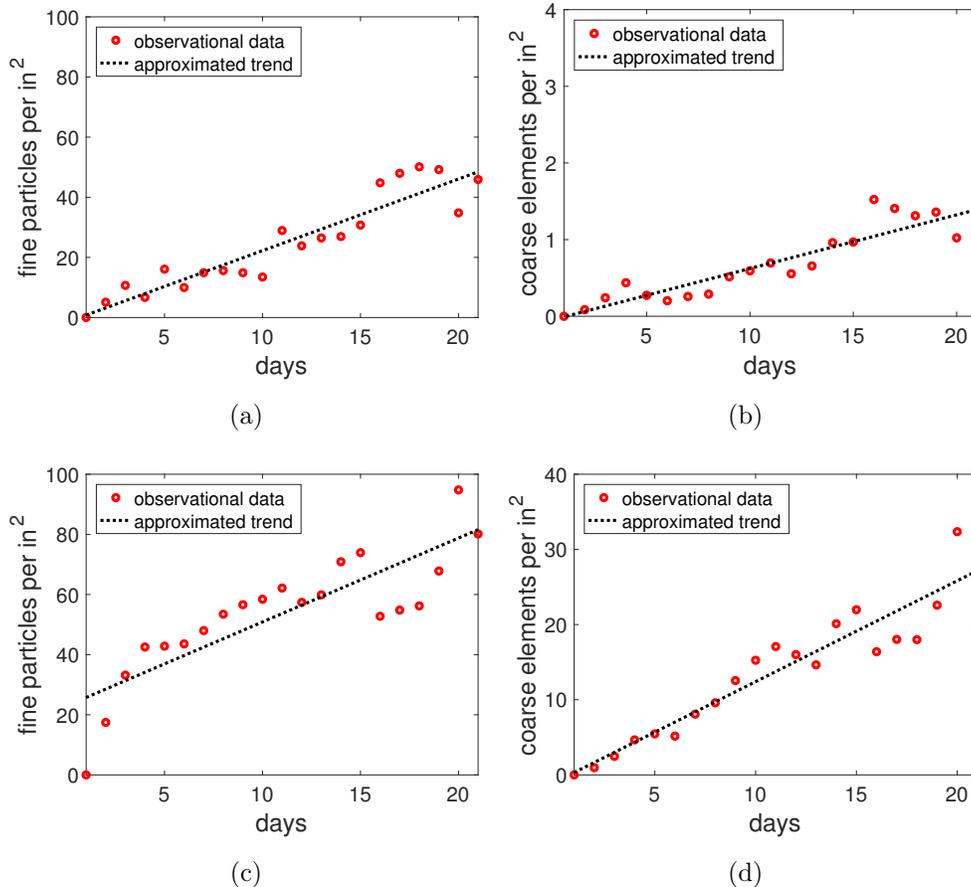
Figure 4.2: Graphs depicting the observational data, along with their respective approximated trends, obtained for the thumbtack (top row) and lego (bottom row) settings. (a) and (c): fine particle density over time. (b) and (d): coarse element density over time.

The results depicted in the graphs presented in Fig. 4.2 indicate that, although the number of fine particles and coarse elements oscillate in relatively short periods of time, they tend to increase in an approximate linear fashion in longer periods of time. We note that such oscillations, which may result from transient changes in air circulation inside a room, are usually perceptually inconspicuous when one takes glances at a dusty scene

within a period of few successive days.

Note that the accumulation rate of fine dust particles in the lego scene depicted in Fig. 4.2(c) is steeper in the first five days than the rate observed in the subsequent days. Since we attempted to keep the environmental conditions constant during our experiments, this initial steeper rate may have been elicited by a slightly stronger level of human presence and activity in the room during the initial five days of the dust accumulation process. This suggests that different periods of dust accumulation within a given setting could be approximated by distinct linear functions in a piecewise fashion. We explore this aspect further in Section 6.

As can also be verified in the graphs presented in Fig. 4.2, the number of fine dust particles marked in a given day was consistently larger than the number of coarse elements. This was expected since some of these elements often result from the aggregation of many fine particles over time. Lastly, the density of particles and coarse elements was considerably higher in the lego scene set in a room characterized by a higher level of human presence and activity. This observation is in agreement with empirical evidence reported in the literature indicating that human skin fragments represent one of the main sources of indoor dust [48].

Chapter 5

Proposed Framework

In this chapter, we describe the proposed texture generation framework, whose design was guided by the qualitative and quantitative aspects addressed in Chapters 3 and 4. Throughout its description, we use vector (bold) notation to represent colours in RGB format. We note, however, that the application of the proposed framework can be extended to other colour formats.

We begin by outlining the general approach adopted for generating textures that represent the accumulation of dust over time. We then introduce the procedures to generate the textures for the opacity film and granularity mask at a given timestep. Lastly, we address the application of the generated textures to the rendering of dusty scenes.

5.1 Overview of Temporal Aspects

While describing our framework, we focus on scenes in which dust naturally accumulates over time according to the level of human presence and activity. In other words, we consider

that the process is not affected by sporadic disturbances (*e.g.*, a wind gust or a mechanical intervention). We discuss how such disturbances could be potentially accounted for in Chapter 6.2.

For a scene with no sporadic disturbances, once a dust particle is placed at a location in a given timestep, one can assume that it will be present at the same location for the entire simulation. Following this assumption, one can represent the accumulation of dust at a given timestep by iterating through each of the preceding timesteps until all desired dust particles have been placed at their respective locations within the scene.

The outcomes of our observational experiments (Chapter 4) indicated that changes in the density of accumulated dust elements over time may be approximated by a linear relationship. Accordingly, we elected to calculate the number of dust elements to be *added* to a target object's surface at a given timestep t (in days) as:

$$p(t) = \begin{cases} A(\alpha + \beta) & t = 1, \\ A\alpha & t > 1, \end{cases} \tag{5.1}$$

where A represents the approximate surface area of the target object in in^2, and the parameters α and β correspond to densities expressed in terms of the number of dust elements per unit of area (in^2). A large value for α would indicate a steeper rate of dust accumulation, which would be appropriate for settings characterized by a high level of human presence and activity. A large value for β would indicate a high initial (after one day) amount of deposited dust elements.

5.2 Opacity Film

As indicated in Chapter 3, the opacity film consists of fine dust particles. We represent the appearance of this film as a texture map (raster file) to be used in the final rendering stage (Chapter 5.4).

To account for the variation in colour that one would expect in a real setting, we consider multiple particles with distinct chromatic attributes falling on the same location at a given timestep. More precisely, we employ a grid-like structure, illustrated in Fig. 5.1, to track the particles at each location, and select a set of colours for them based on the type of environment being considered. For example, from our observations, we found three shades of gray to be sufficient to depict household dust in most cases. In Chapter 6, we examine such choices more closely.
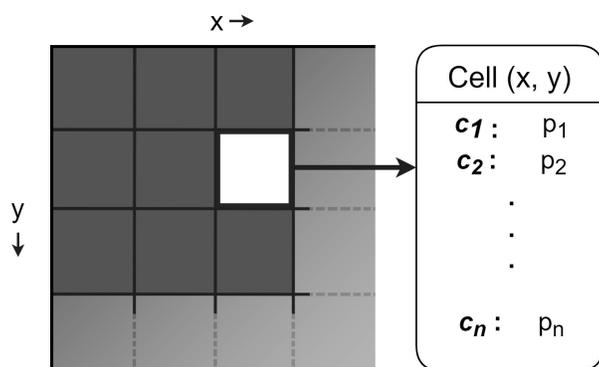
Figure 5.1: Schematic description of the grid-like structure employed during the opacity film generation procedure. For each cell of the grid, we store the colours (labeled c_1 through c_n, from a selected set) assigned to the particles placed within that cell as well as the number of particles (labeled p_1 through p_n) associated with each of the selected colours.

For simplicity, we can select a grid with dimensions equal to those of the desired texture

map, *i.e.*, we can associate each cell with a pixel (on the raster file representing the texture map). Note that it is also feasible to associate a cell with multiple pixels, which could be used to represent larger particles.

We use Eq. 5.1 to calculate the number of fine particles to be added to a target object's surface at any given timestep. Subsequently, for each of these particles, we select a random location within the bounds of the corresponding texture map and one of the colours from the pre-defined colour set. After we determine the grid cell associated with the selected location, we increment the number of particles of the matching colour within that cell. Once all of the particles have been added to the grid, the final particle colour distribution at each cell is employed to calculate the resulting colour of the pixel associated with that cell.

More specifically, to obtain the resulting pixel colour at cell $\{x, y\}$ and timestep t, we use the following sets of equations. We begin by computing the colour $\boldsymbol{u}$ for a given cell as:

$$\boldsymbol{u}_{x,y,t} = \sum_{i=1}^{n} \frac{p_{x,y,t,i}\boldsymbol{c}_i}{p_{x,y,t}}, \tag{5.2}$$

where $p_{x,y,t,i}$ represents the number of particles associated with the ith colour that have accumulated at the given location and time, $p_{x,y,t}$ represents the total particles accumulated at the given location and time, and $\boldsymbol{c}_i$ represents the ith colour.

Afterwards, to adjust the particle opacity, we employ a parameter that represents the adjusted number of particles at a location $\{x, y\}$ in a given timestep t. This parameter, denoted ANP, is computed as:

$$ANP_{x,y,t} = \sum \omega_i p_{x,y,t,i}, \tag{5.3}$$

21

where w_i represents the selected weight of the ith colour. A high weight will result in particles of that colour being more opaque.

Using the parameter ANP, we then compute the relative opacity value, denoted RO, for the cell at a timestep t as:

$$RO_{x,y,t} = \frac{ANP_{x,y,t}}{ANP_{max}} \; \epsilon \; [0,1], \tag{5.4}$$

where the value of ANP_{max} is selected according to the desired appearance. In general, smaller ANP_{max} values will result in an opacity increase for every dust particle in the resulting texture map. This step enables fine control over the overall opacity of the texture map.

Lastly, we calculate the resulting pixel value for our texture map, denoted $\boldsymbol{C_{op}}$, by multiplying our computed colour and opacity values:

$$\boldsymbol{C_{op}}_{x,y,t} = \boldsymbol{u}_{x,y,t} RO_{x,y,t}. \tag{5.5}$$

5.3 Granularity Mask

Similar to the opacity film, we chose to represent the granularity mask as a texture map, and used Eq. 5.1 to calculate the total number of coarse elements to be added to a target object's surface at any given timestep. Unlike the opacity film, the particles in the granularity mask vary not only in terms of their chromatic attributes, but also in terms of their size, shape and orientation. Also, based on our visual inspection of real dusty settings, we assume the coarse dust elements to be opaque and place their constituent particles directly on the texture map.

For each of the coarse elements, we select a random location within the boundaries of the corresponding texture map to represent the center point of the shape. Subsequently, the shape representing the coarse dust element is drawn around this center point. In Fig. 5.2, we provide examples of shapes that can be employed to represent coarse dust elements, whose generation is outlined in Sections 5.3.1 to 5.3.3. The output of the selected shape drawing algorithm at a given timestep t can be stored as:

$$b_{x,y,t} = \begin{cases} 1 & \text{if } \{\text{x, y}\} \text{ is occupied by a coarse element,} \\ 0 & \text{otherwise.} \end{cases} \quad (5.6)$$

We can then calculate the resulting pixel (colour) value at each location $\{x, y\}$ on the texture map representing the granularity mask using:

$$\boldsymbol{C_{g}}_{x,y,t} = \boldsymbol{k}b_{x,y,t}, \quad (5.7)$$

where $\boldsymbol{k}$ is a user-defined colour assigned to the coarse dust elements.
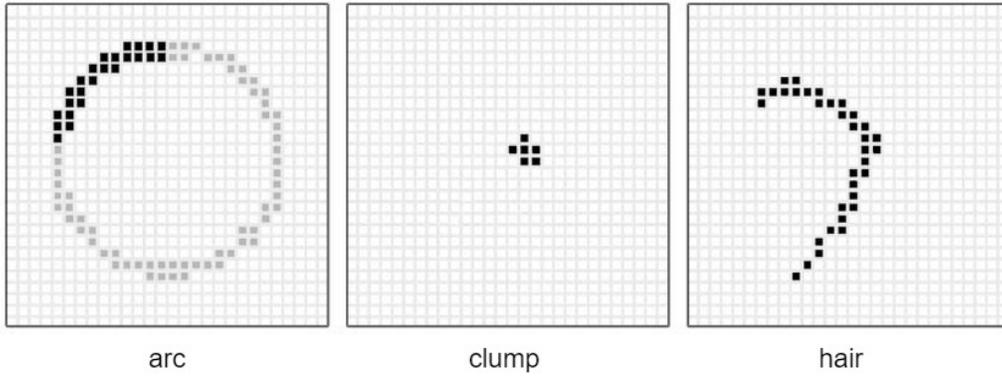
Figure 5.2: Three example shapes (arc, clump, and hair) that may be used by our granularity mask generation procedure to represent coarse dust elements spanning a number of pixels on the corresponding texture map.

23

We note that, similar to the fine particles forming an opacity film, it is possible to have distinct colours assigned to coarse elements forming a granularity mask. In that case, the colour associated with a given pixel could also be specifically or randomly selected from a set of colours previously defined by the user. Again, from our visual inspections of real dusty settings, we found that a single colour is usually sufficient to obtain plausible dust appearance depictions. Thus, although Eq. 5.7 can be modified to account for the assignment of different colours to coarse dust elements, it is provided in this present form for practicality. We illustrate this aspect in Chapter 6.

5.3.1 Arc

For each of the following shape descriptions, we let *rand* represent a random number generator with a range between 0 and 1 (inclusive). Beginning with the arc shape, we describe our implementation with the pseudocode in Algorithm 1. In the proposed procedure, we employ a configuration parameter r to represent the radius (in pixels) of the underlying circle, an angle ϕ to represent the desired arc length, and recommend an integer value n equal to approximately $2\pi r$ to avoid gaps in the rasterized output.

5.3.2 Clump

From our observations, we found that a simple clump implementation consisting of a number of particles p, randomly spread over a rectangular area with height w_y and width w_x, resulted in coarse elements with a believable appearance. We provide the pseudocode for the generation of such clumps in Algorithm 2.

Algorithm 1: Arc generation.

---

x = $rand$ * textureWidth;

y = $rand$ * textureHeight;

γ = $rand$ * (2 * π);

newγ = γ;

for $t = 0\ to\ n$ **do**

    bx = floor(x + r * sin(newγ));

    by = floor(y + r * cos(newγ));

    newγ = newγ + (ϕ / n);

end

Algorithm 2: Clump generation.

---

x = $rand$ * textureWidth;

y = $rand$ * textureHeight;

for $i = 0\ to\ p$ **do**

    bx = x + $rand$ * w_x;

    by = y + $rand$ * w_y;

end

5.3.3 Hair

For the more complex hair elements, we employed three point Bèzier curves, drawn using De Casteljau's procedure [18]. In our application of this procedure, whose pseudocode is outlined in Algorithm 2, we employ a configuration parameter w to represent the maximum width of the generated coarse element, with a larger w value resulting in more noticeable elements (on average).

Algorithm 3: Hair generation.

x = $rand$ * textureWidth;

y = $rand$ * textureHeight;

for $i = 1\ to\ 3$ **do**

    PointsX[i] = x + $rand$ * w;

    PointsY[i] = y + $rand$ * w;

end

for $t = 0\ to\ 1$ **do**

    b = DeCasteljau(PointsX, PointsY)

end

5.4 Applying the Texture Maps to Dusty Scenes

After generating the texture maps for the opacity film and granularity mask, one can proceed to render scenes depicting dusty objects. To realistically represent their appear-

ance using texture maps, however, one should also account for the effects resulting from variations in the viewing angle θ_v and the light incidence angle θ_i (Chapter 3).

To account for the effect of θ_v variations on the dusty appearance at a point $\{x, y\}$ on a target object, we employ a parameter expressed as:

$$V_{x,y} = 1 - (cos\ \theta_{v_{x,y}}), \quad \text{for } 0° \leq \theta_{v_{x,y}} \leq 90°. \tag{5.8}$$

Similarly, to account for the effect of θ_i variations, we employ the following parameter:

$$L_{x,y} = 1 - \frac{1}{2}(cos\ \theta_{i_{x,y}}), \quad \text{for } 0° \leq \theta_{i_{x,y}} \leq 90°. \tag{5.9}$$

We note that, although the presence of dust elements becomes less noticeable when one considers $\theta_i = 0°$ (Chapter 3), they can still be perceived. This aspect can be verified in the photos presented in Fig. 3.1. Hence, we elected to reduce the weight of the $cos\ \theta_i$ term by half based on our empirical observations (Chapter 3).

We can then represent the appearance of the dusty objects in a given scene using the following equations to calculate the colour associated with each point $\{x, y\}$ on their surfaces at a timestep t:

- If $\boldsymbol{C_{g}}_{x,y,y} \neq (0, 0, 0)$, then

$$\boldsymbol{C}_{x,y,t} = \boldsymbol{C_{g}}_{x,y,t} L_{x,y} + \boldsymbol{M}_{x,y}(1 - L_{x,y}), \tag{5.10}$$

  where $\boldsymbol{M}_{x,y}$ represents the colour of a target object's substrate surface at the point $\{x, y\}$.

- Otherwise

$$\boldsymbol{C}_{x,y,t} = \boldsymbol{C_{op}}_{x,y,t} V_{x,y} L_{x,y} + \boldsymbol{M}_{x,y}(1 - V_{x,y} L_{x,y}). \tag{5.11}$$

In Fig. 5.3, we provide a visual representation of this final calculation, showing the texture maps and geometric information used by the proposed framework.
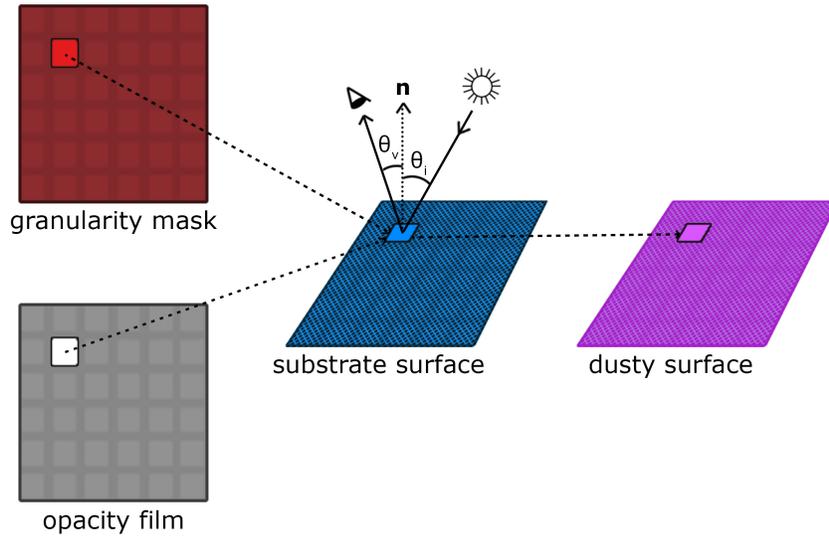
Figure 5.3: A visual representation of the final stage of proposed framework employed in the rendering of a dusty surface. After obtaining the texture maps representing the opacity film and granularity mask, we determine the resulting pixel values for a dusty surface taking into account the viewing and light incidence angles, represented by θ_v and θ_i respectively, and measured with respect to the surface's normal vector $\boldsymbol{n}$.

To further enhance the perceived degree of realism of certain rendered scenes (*e.g.*, depicting curves objects), we found it beneficial to incorporate to the proposed framework a spatial falloff computation step based on the work of Hsu and Wong [26]. For that purpose, we consider a dust source placed directly above a target object, and employ the z-component of the object's normal vector at the point of interest to determine the strength of dust falloff at this point.

More precisely, we begin by mapping the z component to values between 0 and 1, which

will be used to determine the strength of the spatial falloff. We then multiply the resulting mapped value, denoted z_m, by the dust colour value $\boldsymbol{C}_{x,y,t}$, and $(1 - z_m)$ by the material colour value $\boldsymbol{M}_{x,y}$, to obtain the target object's adjusted dusty appearance at the point of interest.

For scenes depicting objects with crevices and/or objects in relatively close proximity (*e.g.*, the block of sticky notes and the lamp shown in Fig. 1.1(c)), we also found it beneficial to take into account ambient occlusion [36]. We note, however, that this procedure can add an additional level of plausibility at the expense of a higher computational cost. Similar to the application of the dust falloff technique mentioned before, one can multiply the resulting ambient occlusion factor (between 0 and 1) by the colour value obtained using Eqs. 5.10 or 5.11.

Chapter 6

Results and Discussion

In this chapter, to showcase the applicability of the proposed framework, we present five test scenes, henceforth referred to as binder clip, wood pieces, teapot, sphere and desktop. We also highlight key aspects associated with the appearance of the depicted dusty objects and indicate possible extensions to the proposed framework.

For all test scenes, we found the following three colour shades (in RGB format) to be sufficient for representing the fine particles forming the opacity films: dark gray (100, 100, 100), medium gray (150, 150, 150) and light gray (200, 200, 200). For the granularity masks, we elected to use a single colour (220, 220, 220) to be assigned to their coarse dust elements.

We considered the placement of each dust element to be equally likely across the entire object, and used a uniform random distribution accordingly. During the generation of the texture maps, we employed a single pixel (on the respective map) to represent a fine particle, and found a resolution of 150 pixels per *in* appropriate for this purpose.

Furthermore, we used ANP_{max} equal to 1.2 in all test scenes, and vary the values as-

signed to the A and w parameters (Table 6.1) used in the generation of their respective texture maps. Lastly, our selection of values for the α and β dust accumulation parameters employed in each scene took into consideration the outcomes of the observational experiments described in Chapter 4. These values, along with the level(s) of human presence and activity associated with each scene, are also provided along with the results presentation.

Scene	A (in^2)	w (pixels)
paper clip	146	30
wood pieces	100	12
teapot	1600	24
sphere	4000	14
desktop	814	36

Table 6.1: Values assigned to the framework parameters A (target surface area) and w (coarse element maximum width) used in the generation of the textures employed in the rendering of the test scenes.

To obtain the images presented this chapter, we implemented the proposed framework in C++, and ran it on a computer with a GTX 1070 GPU and Ryzen 9 3950X CPU. To render the scenes depicted in these images, we used Blender's [16] Cycles path tracing[1] feature (Appendix B), with a custom Shader node setup according to Eqs. 5.9 to 5.11. It is worth mentioning that the generation of the dust textures at 150 ppi for the largest test scene averaged 70 seconds. The rendering of this scene, which is depicted in Fig. 10(f), involved the placement of 1,187,200 fine particles and 551,200 coarse elements.

---

[1]A Monte Carlo method for rendering synthetic scenes while accounting for global illumination effects [30].

31

6.1 Rendered Scenes

Initially, we employed the binder clip scene (Fig. 6.1) to examine the effects of varying viewing and illumination geometries on the appearance of dusty objects, notably after an extended period of dust accumulation. More specifically, we generated images using two distinct values for the viewing and light incidence angles, namely 0° and 80°. To obtain the texture maps, we considered values for the α and β parameters appropriate for representing an environment characterized by a low level of human presence and activity. Specifically, for the opacity film, we used α and β values equal to 2.4 and 0.8 dust elements per in^2, respectively, while for the granularity mask, we used α and β values equal to 0.1 and 0 dust elements per in^2, respectively. Upon visual inspection of the resulting images, one can observe that, as expected (Chapter 3), the opacity film becomes more noticeable as the viewing angle is increased (Figs. 6.1(b) and (d)). Similarly, the combined effects of the opacity film and granularity mask appear most noticeable when the light incidence angle is increased (Fig. 6.1(d)).

We note that dust appearance variations elicited by changes in the illumination and viewing geometries (Fig. 3.1) become more noticeable when one accounts for the distinct roughness levels of dust layers. To further illustrate this aspect, in Fig. 6.2, we presented images obtained for the wood pieces scene using a well-known physically-inspired method [26] as well as images generated using our proposed method, with the opacity film's α and β values set to 2.4 and 0.8 dust elements per in^2, respectively, and the granularity mask's α and β values set to 0.1 and 0 dust elements per in^2, respectively. As can be observed in the images present in Fig. 6.2(c) and (d), changes in the viewing angle have a negligible impact on the appearance of the deposited dust layer when distinct levels of roughness are not taken into account. Conversely, as can be observed in the images depicted in Fig. 6.2(e)

(a)

(b)

(c)

(d)

Figure 6.1: Images of a dusty scene generated considering distinct viewing (θ_v) and light incidence (θ_i) angles measure with respect to the flat surface's normal vector. (a) $\theta_v = 0°$ and $\theta_i = 0°$. (b) $\theta_v = 80°$ and $\theta_i = 0°$. (c) $\theta_v = 0°$ and $\theta_i = 80°$. (d) $\theta_v = 80°$ and $\theta_i = 80°$.

and (f), changes in the viewing angle result in the expected appearance variations when these levels are accounted for.

(a)   (b)

(c)   (d)
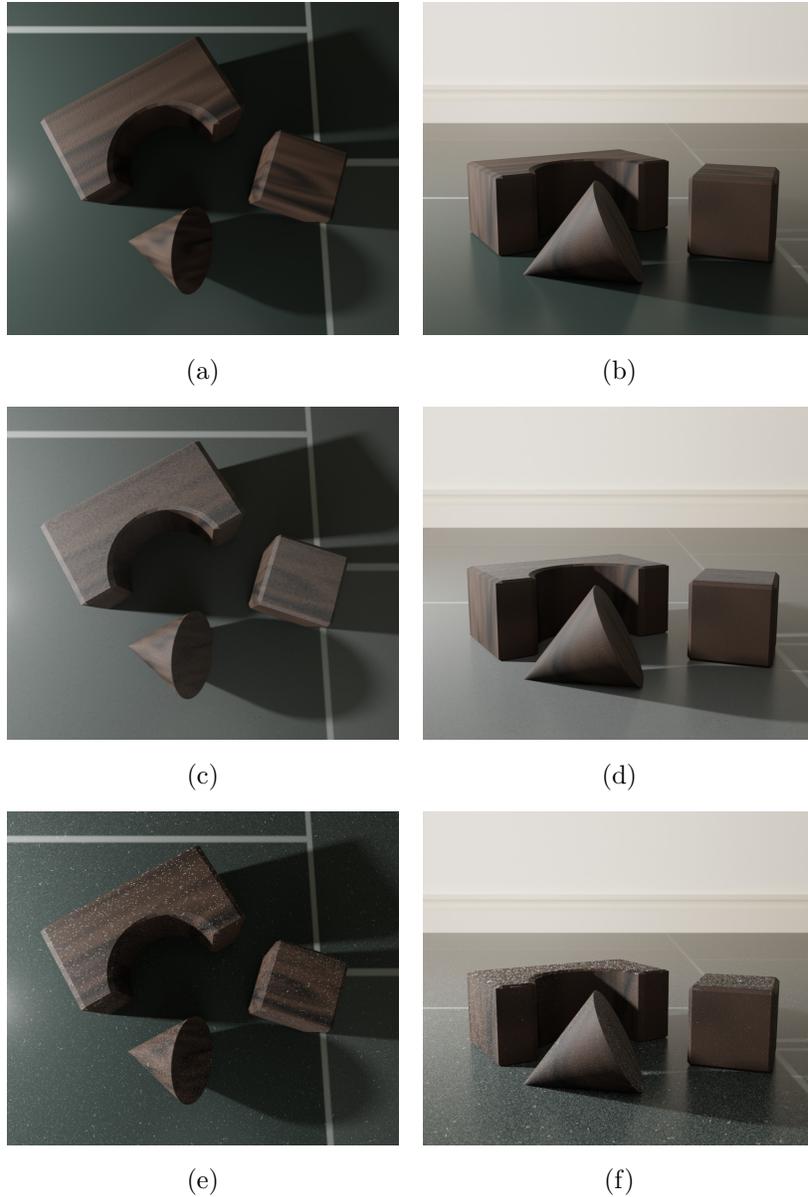

(e)   (f)

Figure 6.2: Images of a scene without (top row) and with accumulated dust. The dust representations in the latter were generated using a well-known physically-inspired method [26] (middle row) and the proposed framework (bottom row). Two distinct viewing (θ_v) angles were considered: $\theta_v = 10°$ (left column) and $\theta_v = 80°$ (right column).

34

We then used the teapot scene (Fig. 6.3) to examine the changes in the accumulation of dust over time in environments with different levels of human presence and activity. The images presented in Figs. 6.3(a)-(d) were rendered considering a low level of human presence and activity. Accordingly, for the opacity film, we used α and β values equal to 2.4 and 0.8 dust elements per in^2, respectively, while for the granularity mask, we used α and β values equal to 0.1 and 0 dust elements per in^2, respectively. The images presented in Figs. 6.3(e)-(h), on the other hand, were rendered considering a high level of human presence and activity. Thus, for the opacity film, we used α and β values equal to 2.8 and 25.7 dust elements per in^2, respectively, while for the granularity mask, we used α and β values equal to 1.3 and 0.3 dust elements per in^2, respectively. In the rendering of the images presented in Figs. 6.3, we also employed the spatial falloff technique outlined in Section 5.4 to improve the appearance of dust accumulated on the teapot's surface. As expected, the accumulation of dust is markedly more noticeable in the images generated considering an environment characterized by a high level of human presence and activity (Figs. 6.3(e)-(h)), particularly with respect to the presence of coarse dust elements.

Recall that the dust accumulation rate can vary in a given environment, notably with respect to the deposition of fine particles, under certain circumstances (Section 4, Fig. 4.2(c)). As we suggested earlier, situations like that could be potentially addressed by applying distinct versions of Eq. 5.1, *i.e.*, with distinct values assigned to the parameters α and β, in a piecewise fashion. To illustrate this possibility, we employed this strategy to the opacity film textures used in the rendering of the sphere scene images (Fig. 6.4). In the initial five days (Fig. 6.4 (a) and (b)), we considered a steeper rate with α and β values equal to 11.1 and 5.1 dust elements per in^2, respectively, while for the remaining days (Fig. 6.4 (c) and (d)), we considered a slower rate with α and β values equal to 2.8 and 25.7 dust elements per in^2, respectively. As for the granularity masks, we used α and β values equal to 1.3 and
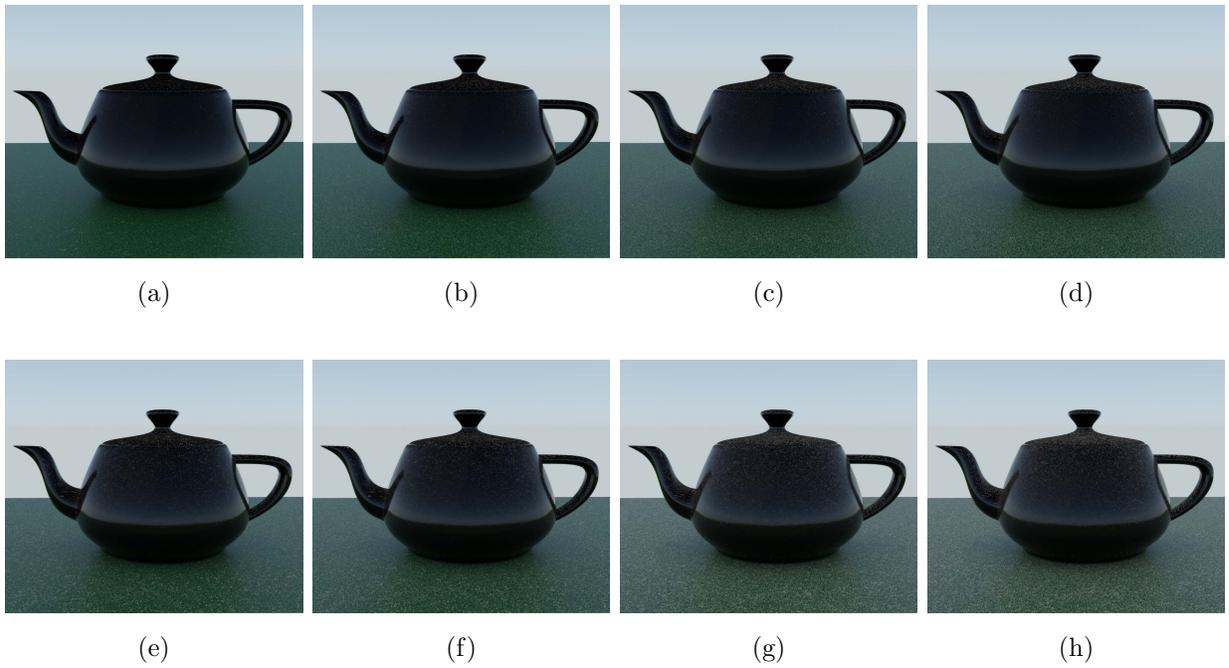
Figure 6.3: Images of a dusty scene depicting dust accumulation over time in environments with low (top row) and high (bottom row) levels of human presence and activity. (a) and (e): after 100 days of dust accumulation. (b) and (f): after 200 days of dust accumulation. (c) and (g): after 300 days of dust accumulation. (d) and (h): after 400 days of dust accumulation.

Figure 6.4: Images of a dusty scene depicting dust accumulation over time considering distinct accumulation rates. (a) and (e): initial state. (b) and (f): after 5 days of dust accumulation. (c) and (g): after 10 days of dust accumulation. (d) and (h): after 100 days of dust accumulation. Fo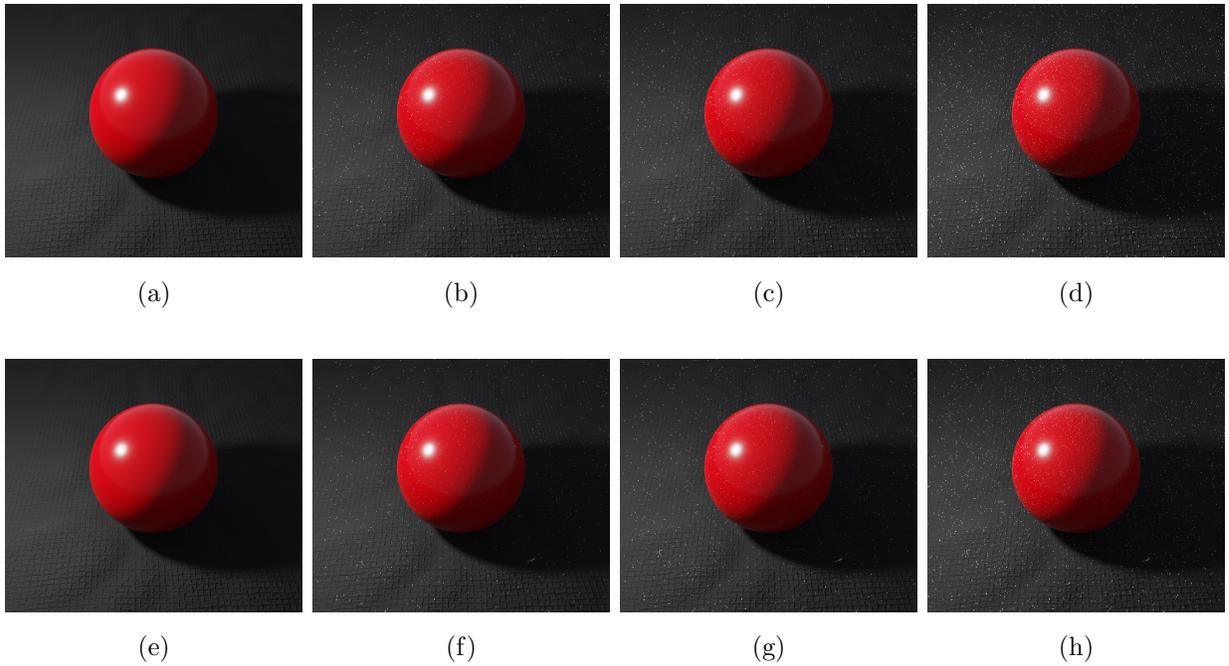r the sequence in the top row, we employed a steeper accumulation rate during the first 5 days, and a lower rate during the remaining days. For the sequence in the bottom row, for comparison purposes, we employed the lower rate for all 20 days.

0.3 dust elements per in^2, respectively. For comparison purposes, we also rendered images obtained considering only the slower accumulation rate for the fine dust particles during the entire dust accumulation period (Fig. 6.4 (e)-(h)). As expected, a piecewise application of Eq. 5.1 can approximate situations like that depicted in Fig. 4.2(c) more closely from a numerical standpoint. Nonetheless, it is also worth noting that, from a visual inspection standpoint, the differences in comparison with a single application of Eq. 5.1 may not be quite significant depending on the values selected for α and β.

Lastly, we employed the desktop scene (Fig. 6.5) to further illustrate the use of the proposed framework in the rendering of believable images depicting changes in the appearance of dusty objects over time. There may be situations in which the level of human presence and activity, combined with distinct air circulation conditions, may affect the accumulation of dust elements in different ways. Thus, for this scene, we elected to consider markedly distinct accumulation rates for the fine particles and coarse dust elements. Accordingly, for the opacity film, we used α and β values equal to 10 and 25.7 dust elements per in^2, respectively, while for the granularity mask, we used α and β values equal to 0.1 and 0 dust elements per in^2, respectively. The provided sequence of images show the objects in an initial ("clean") state (Fig. 6.5(a)), and after dust accumulation periods of 60 days (Fig. 6.5(b)) and 365 days (Fig. 6.5(c)). We also applied the spatial falloff technique in conjunction with ambient occlusion, both mentioned in Chapter 5.4, to represent dust deposits in areas in which the presence of some objects (*e.g.*, the pen and the notebook) can directly affect the accumulation of dust on others (*e.g.*, the desktop). The resulting subtle effects can be observed in the image depicting the removal of the objects on the desktop after 365 days of dust accumulation (Fig. 6.5(d)).

It has been recognized that material appearance perception is a challenging problem for synthetic, pristine objects, even when these objects are subjected to controlled viewing

Figure 6.5: Images illustrating the accumulation of dust in a virtual setting over time. (a) Clean surfaces. (b) Surfaces after 60 days. (c) Surfaces after 365 days. (d) Relatively dust free areas resulting from the removal of objects after 365 days of dust accumulation.

and illumination conditions [11]. Although environmental factors, such as the presence of dust, can provide important visual cues and lead to an increase in the perceived level of realism of these objects, from an optics perspective, they may add yet another layer of complexity to the problem. Certainly, for applications demanding predictive images, it would be appropriate to employ a physically-based framework for the simulation of dust accumulation over time. We remark, however, that the proposed framework was designed using a physically-inspired approach and aiming at the cost-effective generation of plausible images of dusty objects. The sequences of images presented in this chapter indicate that the proposed observation-driven framework can fulfill this objective. Nonetheless we believe that there is still room for enhancements. We briefly address this aspect next.

6.2 Extensions

The proposed framework can be extended in different ways. For conciseness, in the remainder of this chapter, we will outline a few of particular interest, which we intend to explore in our future research.

The first extension would involve providing support for handling sporadic disturbances on the accumulated dust within a given scene. To achieve that, one could define a disturbance function that would remove particles from the grid representing the opacity film, or remove coarse elements from the granularity layer from one timestep to the next. This would mimic processes such as wind gusts or direct manipulations that remove dust elements from a location as they accumulate. It is worth mentioning that the modeling of changes in the appearance of granular materials, such as sand and snow [14, 17, 46], caused by external forces is an active area of research. Although similar concepts could be potentially applied to dust layers, such an undertaking would merit an investigation on its

own right.

A second extension of interest would involve the representation of the particles of the granularity mask as geometric objects rather than as a texture map. We note, however, that although this approach could make the coarse elements more noticeable, it would likely incur significantly higher computational costs.

A third extension could offer an alternative to our use of ambient occlusion. Although the focus of this thesis was not on the optimization of our framework's performance, projects with performance limitations may be unable to rely on ambient occlusion methods for complex objects. We plan to explore alternatives to this method, such as texture maps representing areas of lower dust volume on the target object.

A final extension would focus on streamlining the process of sizing the resulting dusty texture maps. This could involve a method for generating the textures at multiple resolutions for use in conjunction with texture level of detail algorithms [2, 20]. This would be beneficial for scenes involving a large number of objects, where there may be a prohibitive amount of work involved in manually determining the correct texture file sizes to maintain a consistent scale for the simulated dust particles.

Chapter 7

Conclusion and Future Work

We have proposed a physically-inspired framework for the generation of textures depicting temporal variations on dust layers covering objects composing virtual settings. Its design is based on the representation of this ubiquitous material using two abstract levels of roughness, referred to as the opacity film and the granularity mask. The texture maps associated with these two representations are then combined during rendering taking into account the viewing and illumination geometries employed to obtain the target materials' depictions.

In this thesis, we have also described observational experiments that provided the insights leading to the proposed representation of dust layers. Moreover, these experiments allowed us to gather qualitative and quantitative information about time-dependent changes on these layers, particularly due to different levels of human presence and activity in the environments under inspection. This information was then incorporated into our texture generation algorithms. The rendered sequences presented to illustrate the applicability of the proposed framework indicate that it can be effectively employed to obtain

plausible depictions of changes in the appearance of dusty objects over time.

The perceived appearance of real materials is often affected by environmental factors. The accumulation of dust over time is among the most prevalent of these factors. Accordingly, a number of works based on different physically-based and physically-inspired approaches have been proposed to account for these factors and, consequently, increase the degree of realism of synthetic scenes. Viewed in this context, the outcomes of this research can be seen as tangible contributions to these efforts, notably with respect to the generation of believable images of dusty objects at a relatively low computational cost.

As future work, besides addressing the aforementioned possible extensions to the proposed framework, we plan to extend our observational experiments to a wider variety of environments and dust compositions (*e.g.*, dust originating from wind-transported sand). We also intend to investigate alternatives to streamline the application of dust textures in complex scenes, and explore their use in conjunction with first-principles material appearance models in order to enhance the fidelity to cost ratio of predictive renderings of dusty objects. The development of such models coupled with the simulation of external factors, such as static electricity and friction, affecting dust deposition patterns would also represent interesting avenues for future research in this area.

References

[1] S. Adachi, H. Ukaji, T. Kosaka, and S. Morishima. Real-time dust rendering by parametric shell texture synthesis. In *ACM SIGGRAPH 2013 Posters*, July 2013. Article 104. 7

[2] T. Akenine-Möller, J. Nilsson, M. Andersson, C. Barré-Brisebois, R. Toth, and T. Karras. Texture level of detail strategies for real-time ray tracing. In E Haines and T. Akenine-Möller, editors, *Ray Tracing Gems*, pages 321–345, New York, NY, USA, 2019. Apress Media LCC. Chapter 20. 41

[3] G. Ashraf and W. K. Cheong. Dust and water splashing models for hopping figures. *J. Visual Comp. Animat.*, 10(4):193–213, 1999. 5, 7

[4] J.M. Bajo, C. Delrieux, and G. Patow. Physically-inspired technique for modeling wet absorbent materials. *The Visual Computer*, 37:2053–2068, 2021. 3

[5] D. Bandeira and M. Walter. Highlights on weathering effects. *Visual Comput.*, 26(6):965–974, 2010. 1

[6] R. Baravalle, L. Scandolo, C. Delrieux C.G. Bauza, and E. Eisemann. Realistic modeling of porous materials. *Comp. Anim. Virtual Worlds*, 28:1–13, 2016.

[7] N. Bell, Y. Yu, and P. J. Mucha. Particle-based simulation of granular materials. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 77–86, 2005. 5

[8] J. F. Blinn. Light reflection functions for simulation of clouds and dusty surfaces. *Comp. Graph.*, 16(3):21–29, 1982. 8

[9] R.L. Burden and J. Douglas. *Numerical Analysis*. PWS-KENT Publishing Company, Boston, USA, forth edition, 1988. 15

[10] D. Carr and J. Geyer. A study of simulation and representation methods of dust clouds for real-time graphics applications. In *Proceedings of the Second Australian Undergraduate Students' Computing Conference*, page 17. Melbourne: Australian Undergraduate Students' Computing Conference, 2004. 5

[11] B. Chen, C. Wang, M. Piovarci, H.P. Seidel, P. Didyk, K. Myszkowski, and A. Serrano. The effect of geometry and illumination on appearance perception of different material categories. *Visual Comput.*, 37:2975–2987, 2021. 40

[12] J. X. Chen and X. Fu. Integrating physics-based computing and visualization: modeling dust behavior. *Computing in Science and Engineering*, 1(1):12, 1999. 5

[13] J. X. Chen, X. Fu, and E. J. Wegman. Real-time simulation of dust behavior generated by a fast traveling vehicle. *ACM Transactions on Modeling and Computer Simulation*, 9(2):81–104, 1999. 5

[14] X. Chen and Y. Zhu. Real-time simulation of vehicle tracks on soft terrain. In *9th International Symposium on Visual Computing*, pages 437–447, Crete, Greece, July 2013. 40

[15] Y. Chen, L. Xia, T. T. Wong, X. Tong, H. Bao, B. Guo, and H. Y. Shum. Visual simulation of weathering by γ-ton tracing. *ACM T. Graphic.*, 24(3):1127–1133, 2005. 6, 7

[16] Blender Online Community. *Blender - a 3D modelling and rendering package.* Blender Foundation, Stichting Blender Foundation, Amsterdam, 2020. 31, 54

[17] F. Dagenais, J. Gagnon, and E. Paquette. An efficient layered simulation workflow for snow imprints. *The Visual Computer*, 32:881–890, 2016. 40

[18] G. Farin. Algorithms for rational Bèzier curves. *Computer-Aided Design*, 15(2):73–77, 1983. 26

[19] Paul Fearing. Computer modelling of fallen snow. In *SIGGRAPH, Annual Conference Series*, pages 37–46, New Orleans, LA, USA, July 2000. 5

[20] James D Foley, Foley Dan Van, Andries Van Dam, Steven K Feiner, John F Hughes, and J Hughes. *Computer Graphics: Principles and Practice.* Addison-Wesley Professional, Reading, MA, USA, 3rd edition, 1996. 7, 41

[21] R.C. Gonzalez, R.E. Woods, and S.L. Eddins. *Digital image processing using MATLAB.* Gatesmark Publishing, Knoxville, TN, USA, 3rd edition, 2020. 52

[22] D.P. Greenberg, J. Arvo, E. Lafortune, K.E. Torrance, J.A. Ferwerda, B. Walter, B. Trumbore, P. Shirley, S. Pattanaik, and S. Foo. A framework for realistic image synthesis. In *SIGGRAPH, Annual Conference Series*, pages 477–494, 1997. 3

[23] J. Guo and J. G. Pan. Real-time simulating and rendering of layered dust. *Visual Comput.*, 30(6-8):797–807, 2014. 2, 6, 8

[24] D. Hanselman and B. Littlefield. *Mastering MATLAB 6 A Comprehensive Tutorial and Reference.* Prentice Hall, Upper Saddle River, NJ, 2001. 15

[25] D. W. Hoock Jr. Modeling time-dependent obscuration for simulated imaging of dust and smoke clouds. *P. Soc. Photo-opt. Ins.*, 1486:164–175, 1991. 7

[26] S. C. Hsu and T. T. Wong. Simulating dust accumulation. *IEEE Comput. Graph.*, 15(1):18–22, 1995. ix, 2, 5, 6, 7, 28, 32, 34

[27] T. Imagire, H. Johan, and T. Nishita. A fast method for simulating destruction and the generated dust and debris. *The Visual Computer*, 25(5-7):719–727, 2009. 5

[28] Adobe Inc. Photoshop. 51

[29] D.B. Judd and G. Wyszecki. *Color in Business, Science and Industry.* John Wiley & Sons, New York, third edition, 1975. 8

[30] James T. Kajiya. The rendering equation. In *Computer Graphics*, pages 143–150, 1986. 31

[31] B. W. Kimmel, G. V. G. Baranoski, T. F. Chen, D. Yim, and E. Miranda. Spectral appearance changes induced by light exposure. *ACM T. Graphic.*, 32(1):10:1–13, 2013. 1

[32] B.W. Kimmel and G.V.G. Baranoski. A novel approach for simulating light interaction with particulate materials: application to the modeling of sand spectral properties. *Opt. Express*, 15(15):9755–9777, 2007. 5

[33] B.W. Kimmel and G.V.G. Baranoski. Simulating the appearance of sandy landscapes. *Comput. Graphic.*, 34(4):441–448, 2010. 5

[34] P. Kubelka and F. Munk. Ein beitrag zur optik der farbanstriche. *Zurich Tech. Physik*, 12:543, 1931. 8

[35] J. Lu, A. S. Georghiades, A. Glaser, and H. Wu. Context-aware textures. *ACM T. Graphic.*, 26(1):1–22, 2007. 1, 6

[36] A. Méndez-Feliu and M. Sbert. From obscurances to ambient occlusion: A survey. *Visual Comput.*, 25(2):181–196, 2009. 29

[37] J. Meng, M. Papas, R. Habel, C. Dachsbacher, S. Marschner, M. H. Gross, and W. Jarosz. Multi-scale modeling and rendering of granular materials. *ACM T. Graphic.*, 34(4):1–13, 2015. 5

[38] S. Merillou and D. Ghazanfarpour. A survey of aging and weathering phenomena in computer graphics. *Comput. Graphic.*, 32(2):159–174, 2008. 1

[39] G. Miller. Efficient algorithms for local and global accessibility shading. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, pages 319–326, 1994. 6

[40] I. Munñoz-Pandiella, C. Bosch, N. Mérillou, G. Patow ans S. Mérillou, and X. Pueyo. Urban weathering: Interactive rendering of polluted cities. *IEEE Transactions on Visualization and Computer Graphics*, 24(12):3239–3252, 2018. 3

[41] E. Ofek, E. Shilat, A. Rappoport, and M. Werman. Multiresolution textures from image sequences. *IEEE Comput. Graph.*, 2(2):18–29, 1997. 7

[42] K. Perlin. An image synthesizer. *Comp. Graph.*, 19(3):287–296, 1985. 7

[43] R.L.C. Santos and G.V.G. Baranoski. Observation-driven generation of texture maps depicting dust accumulation over time. *Vis. Comput.*, 2022 (in press). 3

[44] M. Schwärzler and M. Wimmer. Rendering imperfections: Dust, scratches, aging,.. Technical Report TR-186-2-07-09, Vienna University of Technology, Austria, 2007. 1

[45] S. Shahidi. Sal weathering of brick walls. In *International Conference on Computer Graphics Theory and Applications*, pages 7–15, Rome, Italy, February 2012. 3

[46] R.W. Summer, J.F. O'Brien, and J.K. Hodgins. Animating sand, mud and snow. pages 125–132, Vancouver, BC, Canada, June 1999. 40

[47] B. Sun, K. Sunkavalli, R. Ramamoorthi, P. N. Belhumeur, and S. K. Nayar. Time-varying BRDFs. *IEEE T. Vis. Comput. Gr.*, 13(3):595–609, 2007. 2, 8

[48] J. E. M. H. van Bronswijk. *House dust biology for allergists, acarologists and mycologists.* NIB Publishers, Zeist, The Netherlands, 1976. 1, 17

[49] C. J. Weshchler, S. P. Kelty, and J. E. Lingousky. The effect of building fan operation on indoor-outdoor dust relationships. *J. Air Pol. Control Assoc.*, 33(6):624–629, 1983. 1

[50] H.K. Yuen, J. Princen, J. Illingworth, and J. Kittler. Comparative study of Hough transform methods for circle finding. *Image Vision Comput.*, 8(1):71–77, 1990. 52

[51] J. Yun, M. Son, B. Choi, T. Kim, and S. Yoon. Physically inspired, interactive lightning generation. *Comp. Anim. Virtual Worlds*, 28:1–10, 2017. 3

APPENDICES

Appendix A

Gathering Dust Accumulation Data from Photos

In this Appendix, we provide a concise of overview of the standard tools used to gathered the dust accumulation from our observational experiments described in Section 4. We remark that our observational experiments and the analysis of their outcomes were intended to provide approximations (rather than precise readings) to guide the development of the proposed physically-inspired framework.

A.1 Marking Dust Within the Photos

The first step to gather the dust accumulation data involved the generation of raster files consisting of small ellipses in the locations of each of the dust particles. Accordingly, the images were processed using an image editing software (Adobe Photoshop [28]), and a transparent layer was created above each photo. On this new layer, areas of the underlying

photo containing dust particles were marked with an ellipse. The stages of this process are illustrated in Fig. A.1.

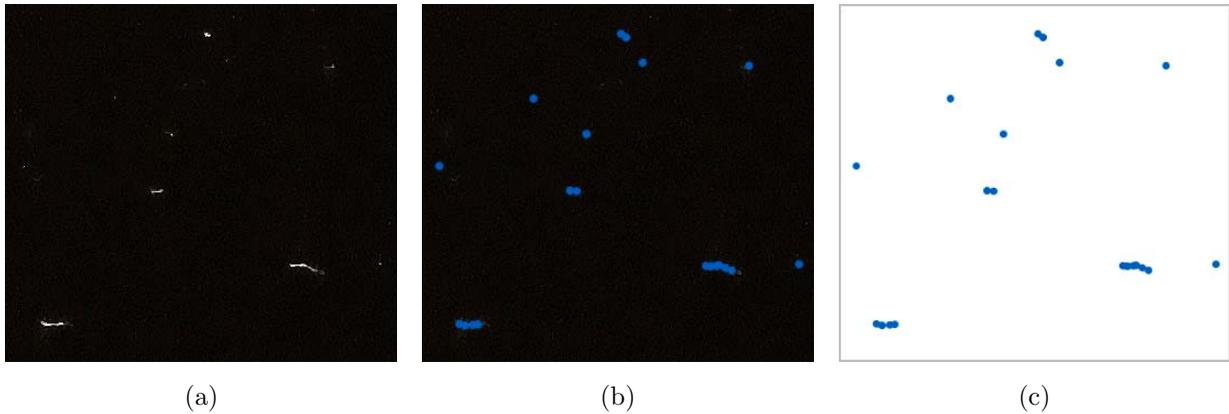| (a) | (b) | (c) |

Figure A.1: A cropped example of the stages of the particle marking phases of the dust accumulation experiments described in Chapter 4. (a) A small section of a photo from our accumulation experiments. (b) Photo with ellipses overlayed where particles are seen. (c) Resulting raster file to be processed in MATLAB.

A.2   Using MATLAB to Count the Markings

With the particles converted into ellipses, we were able to count the markings within the final images using the *imfindcircles* MATLAB function [21], which relies on a circular Hough transform [50]. Our usage of this function is shown in Fig. A.2.

```matlab
for n=1:20
    ImageDataParticles{n} = imread(sprintf(`lego/day%d_particles.jpg',n));
    ParticleCount{n} = size(imfindcircles(ImageData{n}, [3 8], `ObjectPolarity',
       `dark', `Sensitivity', 0.98), 1);
end
```

Figure A.2: MATLAB code fragment employed to convert photos containing elliptical dust markings into a count of these markings.

Using the code provided in Fig. A.2, we gathered the fine particle and coarse element counts for the two scenes described in Chapter 4. Once gathered, we relied on MATLAB's graphing capabilities to create Fig.4.2(a)-(d).

Appendix B

Configuring Cycles Shader Nodes within Blender

For the purposes of this thesis, we relied on Blender's [16] rendering tool *Cycles* to render the final images. *Cycles* is Blender's path tracing rendering option, and it supports a level of customization that enabled us to easily implement our framework.

A key feature of *Cycles* that we relied on to accurately implement our rendering algorithm was the Shader node editor. This feature allows the user to compose various 'nodes' representing different rendering operations. In Fig. B.1, we illustrate the combination of Shader nodes that we found most helpful for the implemention of the proposed framework.

As shown in Fig. B.1, there are two basic Blender nodes that we relied on to render our images. The first is the Math node set to multiply, which simply represents a scalar multiplication operation. The second is a Mix node, which takes two shaders as inputs and one scalar input value ranging between 0 and 1, and outputs a combination of the two shaders weighted by the scalar value.
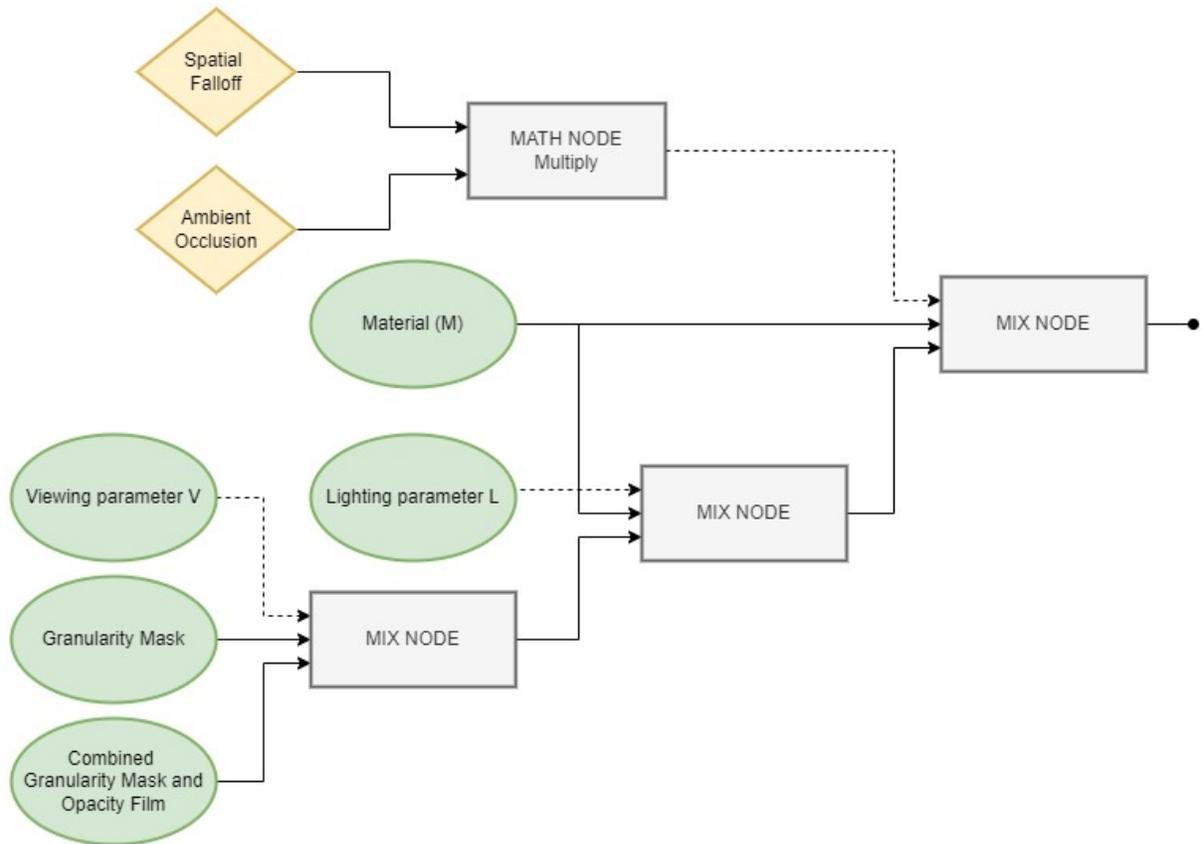
Figure B.1: A diagram showing the graph of Shader nodes used to implement our framework. Green nodes represent basic Shader nodes found in Blender, elliptical nodes represent the major concepts defined in Chapter 5, while diamond-shaped nodes represent optional parameters. Outputs illustrated with solid lines represent shader values, while dotted lines represent scalar values.

Alphabetical Index