Corrosion Rendering: Fusing Simulation and Photo-texturing

Nisha Jain
IIT Delhi
Hauz Khas, Delhi
110016, India
nisha@cse.iitd.ac.in

Prem Kalra
IIT Delhi
Hauz Khas, Delhi
110016, India
pkalra@cse.iitd.ac.in

Subodh Kumar
IIT Delhi
Hauz Khas, Delhi
110016, India
subodh@cse.iitd.ac.in

Abstract

We present a technique for realistic rendering of corroded objects. We employ a physio-chemically based stochastic model to determine the deterioration level of different points on an object, given its material characteristics and the vigor of the environment. Guided by values from the ISO standard, our model predicts shape degradation. This shape degradation is then applied to the object in the form of surface displacements and weathered appearance. The appearance degradation is hard to physically model accurately due to its dependence on a large number of unknown parameters as well as its high sensitivity to errors in modeling them. Hence, we instead sample from photographs of real objects to generate similar appearance for the rendered surface, but consistent with the simulated corrosion levels. We demonstrate our technique using several simulation results as well as different input photographs. We also evaluate the fidelity of the generated output to the simulation as well as to the sample texture patterns and validate our work with the help of data published in the corrosion literature. Our framework is generic and can be extended to a variety of corrosion scenarios. Ours is an important step towards predictive analysis of material loss and weathering phenomena for real objects.

Keywords

Corrosion rendering

1 INTRODUCTION

Corrosion is a stochastic process impacted by non-linear combination of factors like material property, environment, exposure time, etc. The complexity of corrosion makes it difficult to accurately predict the complete state of weathering objects, or even to measure all the causal factors. Simulation and rendering of weathered 3D objects is necessary for many application, games, movies, aesthetic design, and even unweathering of already weathered objects. Predictive models of weathering can also be used to estimate structural damage. We extend the model presented in [20] to general weathering and demonstrate its effectiveness in estimating corrosion. Such estimation of the physio-chemical state of an object undergoing corrosion has applications across diverse fields such as arts, engineering, aerospace, etc. Predictive analysis for an object undergoing corrosion is an important part of taking protective measures against both appearance and shape damages.

Corrosion can be decomposed into two basic building blocks including acceleration process due to an auto-catalytic chemical reaction followed by deceleration due to physical formation of semi-passive porous layer. Corrosion as a process results in both state change resulting in aesthetic degradation as well as material loss resulting in structural damage of the exposed layer. We model corrosion as a stochastic process influenced by the probabilities dependent on material and environmental conditions. The simulation works on a voxelization of the input object undergoing corrosion. In addition to our model being agnostic to the object size, shape intricacies and voxel resolution, the in-built continuous functions ensure that the model structure itself is generic and independent of specific material type or environment corrosivity, allowing them to be provided as inputs. The model allows each voxel to progressively reach higher corrosion states before being ultimately removed, exposing inner layers. To replicate actual corrosion process we also ensure the additive effect of highly corroded neighborhood and decelerating impact caused by creation of semi-porous layer of inert oxides.

Because of applications of corrosion study across domains, there has been significant effort to measure and simulate corrosion reported in the literature [14]. How-

 ISBN 978-80-86943-65-7

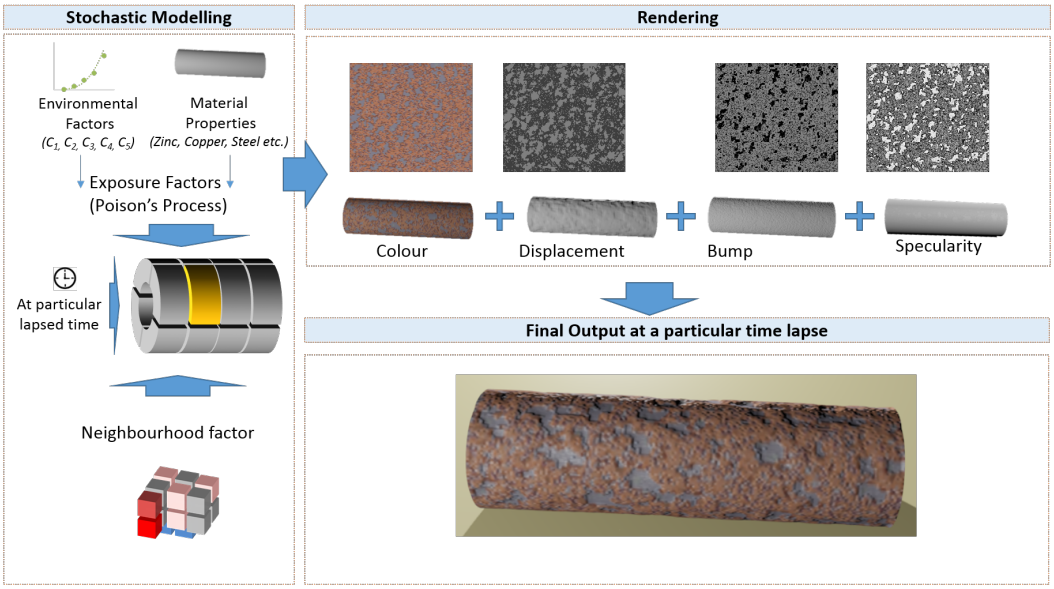


Figure 1: Simulation and Rendering Pipeline : The first block is the stochastic model, which includes voxelization and the corrosion simulation. The second block is the rendering pipeline, which generates different maps and combines them to create the final rendered corroded object.

ever, an accurate prediction of the shape of a corroded object has remained elusive. On the other hand, effects like failure time, pit depth, material loss etc., over time have been measured and simulated [21]. We extend the pitting corrosion model of Jain et al. [20] to compute the shape of the object at any given time. The simulation is driven by measured material loss included in ISO standard [14]. We demonstrate that data from the ISO standard can be used to effectively guide the stochastic parameters of the model to ensure that the predicted material loss closely follows the expected material loss. At the same time, our robust rendering framework ensures that the resulting shapes are realistic. The simulation additionally produces the degree of corrosion (normalized between 0 and 1) for every point on the surface at a given time interval. In order to create a realistic rendering, the reflectance properties of each corroded point needs to be derived.

Optical properties of the surface can vary in largely unpredictable fashion due to corrosion. Mostly in game development tools the textures are hand modeled and do not incorporate the actual corrosion process. Also, coexistence of multiple dynamic states due to the material's interaction with the environment results in complex and dynamic appearance, which a real photograph can best capture. Hence, instead of simulating it physio-chemically, we sample the appearance from parts of real photographs of other corroded objects, assuming that they exhibit the appearance of a variety of differently – some slight, others significant – corroded areas. Using these areas as examples then, we generate the appearance of the simulated object. We first normalize the photographs and subtract the effect of lighting to derive the underlying albedo [5]. Similar to [24], we then allow a user to mark corrosion degrees for a few pixels in the albedo map. The appearance manifold technique of [24] is then employed to derive the corrosion degree of all the un-marked pixels. This albedo map is then used to create reflectance values for the corroded material. This novel integration of physical simulation and appearance synthesis produces plausible results.

Ours is a more holistic approach than, say, predicting only material thickness or material loss, given a particular combination of factors (corrosion level and time). It predicts actual shape changes as well as the degree of corrosion, before imposing the reflective properties of the corroded material that is consistent with the corrosion.

2 RELATED WORK

Corroded objects have compound construction and their structure dynamically changes under the effect of varied physio-chemical conditions. These physio-chemical conditions are influenced by the interaction of changing environment and the deteriorating object. This results in making corrosion a complicated stochastic process where even a given material may deteriorate differently given its state and environmental conditions.

Given its relevance in varied fields including civil engineering, aerospace, heavy industries and art, extensive work has been done to understand corrosion phenomena [10] [16]. Though these work try to predict physical loss of material, not much work has been done on es-

timating structural in conjunction with aesthetic degradation.

Corrosion damages can be broadly categorized into localized and uniform corrosion [23]. Pitting corrosion is a severe form of localized corrosion that causes small cavities on the surface. Jain et al. present a model for pitting corrosion [20], which we generalize in this paper. Their work is limited to pitting corrosion with no notion of real time, resulting in lack of comprehensive predictive modeling. Our basic model uses there model as core but is tightly coupled to real time, thus opening the possibility of predictive analysis of corrosion for major materials including steel, copper and zinc.

We simulate uniform corrosion to validate our results given the lack of weight loss data for other forms of corrosion. Uniform corrosion is a generic form of corrosion where the attack proceeds evenly throughout the surface. There are a few models for simulation of general corrosion. Guessasma et al. [11] simulate the corrosion phenomena under potentiostaic conditions to the behavior of the process. They generate the simulation on a 3D grid and study the current density and exposed area. They simulate the corrosion process but do not produce real shape degradations.

There have been other studies on corrosion of different materials [9] [22]. It is observed that different materials behave differently and the subsequent surface degradation is discrete. Mérillou et al. [19] predict the changes due to corrosion with time. They best fit the ISO standard [14] [15] experimental data for steel and use the weight loss information to remove material from a three dimensional corrosion map grid. They do not simulate the actual process on a three dimensional object. Also they modify the BRDF model to affect the porosity and roughness of the surface in the final shading model. This is not based on any physical evidence. Our model gathers all the degradation effects from the actual stochastic simulation of the corrosion phenomena.

General corrosion analysis consists of predicting the extent of attack with time, under specific atmospheric conditions. ISO standards [15] list the weight loss in grams per meter square for specific materials under certain atmospheric conditions over time. They categorize environment corrosivity levels and catalog material weight loss under different conditions. We validate our results by generating similar weight loss from objects in our simulation.

Realistic rendering of corroded objects is vital in the field of computer graphics [7] [6][8]. Wang et al. [24] produce the optical appearance of a weathered surface by generating an "appearance manifold". They apply appearance sampling, but only approximately follow the general trend of corrosion variance across the surface. Also they use a 7D feature vector which is cumbersome to gather. They apply stochastic models to deteriorate color but are unrelated to the actual corrosion process. Bandeira et al. [1] create weathering effects based on chroma and luminance values. They introduce the concept of appearance maps similar to [24] but they do not address rough geometry variations. Their method works only on images and they do not present any three dimensional procedure. Hwang et al. [13] propose a method for creating a weathering gallery based on time dependent appearance manifolds(TDAMs). They create these TDAMs from sample videos clips. The major difficulty in their proposed method is collection of sample videos which covers presents a good weathering phenomena with least change in camera position and illumination characteristic.

Textures have also been used [26] to capture realistic surface effects in this domain, γ-ton tracing [3], is a particle based surface-centric model of corrosion. It does not handle severe geometry degradation that follows corrosion statistics. Clément et al. [4] generate aging textures by taking as input a target aging mask from the user. They employ elimination and reproduction techniques for producing the final texture. Their process is simple but requires a lot of user intervention and thus the user needs significant domain knowledge to achieve realistic results. Wojtan et al. [25] generate weathering by increasing or decreasing the surface of the object but the results produced are quite synthetic and do not appear real. Kamata et al. [17] describes a model for texture regeneration of peeling of preservative coatings of surfaces. They show the effects of peeling using geometric and environmental maps. Their model produces this effect by striking the surface with water drops and calculating the amount of accumulated droplets on the surface which leads to peeling. Neither do they incorporate the actual corrosion phenomena nor do they have a real time assessment of the corrosion process.

We propose a time variant voxel-based model to simulate and render generic corrosion on different materials. Our model produces geometric results based on the stochastic simulation and color changes based on texture synthesis, onto the object to produce photo-realistic effects. We validate our results based on ISO standard [15]. The validation is strengthened by observing the weight loss from different materials at different time-steps, which is similar to the standard. We demonstrate surface degradation with respect to time for different materials.

3 MODELING OF CORROSION

Corrosion is the deterioration of an object with specific material properties under dynamic environmental condition. Due to the dynamic change in intensity as well as the number of factors influencing the shape and aes-

thetics of an object undergoing corrosion, it is hard to accurately predict object state with elapsed time.

Jain et al. [20] simulate *pitting* corrosion as a two-step stochastic process. The first step nucleates pits and the second step grows them. The division of simulation into these two steps is based on the physio-chemical nature of the pitting corrosion process. Their model is qualitative in nature and does not account for the real weight loss in the object after a given time elapses. There is no real-time scale. Their stochastic model of corrosion growth, however, is inspired by the physio-chemistry of corrosion process. We improve their model in this paper and apply it to general corrosion with real time scale.

In order to introduce the notion of real time-scale, we take recourse to the standardized observations of the corrosion phenomena [15] to predict the actual weight loss in a given time for a particular material in a type of environment. We employ the stochastic aging process to obtain a single step corrosion model and modulate it to produce the observed weight losses in a given time period as cataloged in the ISO standard.

3.1 Corrosion Rate Categorization

Atmospheric condition such as rural, urban, industrial, marine, chemical, etc., varies and is one of the major contributors to corrosion. The complex inter-play of various factors such as metallic properties, environmental factors and operating conditions make accurate prediction of detailed corrosion behavior of different materials nearly impossible to track. The ISO standard [15] classifies the character of corrosion attack into five "corrosivity" levels as shown in Table 1. Corrosivity is a measure of the ability of the atmosphere to cause corrosion in a given corrosion system. For different materials (e.g., Steel, Copper, Aluminium, etc.) exposed to each class, ISO lists the average weight lost per unit surface area in each year of exposure, We call this the "corrosion rate."

Corrosivity	1 year	5 years	10 years
Very Low (C1)	10	23	33
Low (C2)	200	464	668
Medium (C3)	400	928	1334
High (C4)	650	1508	2167
Very High (C5)	1500	3480	5001

Table 1: Corrosion rate for steel for different standardized corrosivities [15] for different exposure times

Table 1 shows the corrosion rate in grams per m^2 of steel when exposed to different standardized corrosivities [15] for different exposure times. By fitting a function to the table entries, we can compute the amount of material lost in different environment types (namely, corrosivity categories C1, C2, C3, C4, C5 and intermediate levels) during different time periods. The corrosion standard has similar corrosion rate tabulation for different materials. We best fit the values in these tables to generate a function that gives weight loss per unit area for a particular material, given the atmospheric corrosivity, for a given elapsed time t. The following form fits all materials and conditions:

Material	k	n	m
Steel	13.84	2.9621	0.5257
Copper	0.97416	2.4492	0.67018
Zinc	0.7	2.7355	0.8169

Table 2: Constants for different materials for Equation 1

$$\Omega = k * C^n * t^m \tag{1}$$

where, Ω is the corroded weight per unit area, C is the corrosivity level and k, n, and m are constants whose values are given in Table 2 (with different values for different materials). Our stochastic corrosion model is now guided by Equation 1.

3.2 Model Building Blocks

We start with a voxelization of the corroding object as in [20]. As a result of this parametric voxelization, the voxels in the object space are warped cubes [2]. Voxels account for the geometric structure of a solid object. In this paper, we incorporate physical properties of the voxel, i.e., each voxel is assigned a mass based on its volume and material density.

We follow the framework of [20] as the core of our model, which we describe first. Each voxel has six face neighbors. A voxel with at least one face exposed is a boundary voxel. The simulation processes only the boundary voxels. Each voxel maintains a corrosion level (ψ), which accounts for the actual decay state of the voxel. It ranges from 0 to 1. At level 0, an exposed voxel is fresh material. At 1, the voxel is fully corroded and removed from the simulation, exposing voxels behind it. At each step of the simulation, the corrosion level of each exposed voxel is incremented by a constant factor δ based on a probability distribution function and the average level of corrosion around it.

We borrow the following intrinsic parameters of the simulation employed by [20].

- ν: It controls the influence of the immediate neighborhood. This aggravates the corrosion due to the flow of anions in the medium.
- ω: A constant value that accounts for material strength resistance to get corroded. When ψ at a point exceeds ω, the voxel is removed.
- δ: This is the factor by which ψ of a voxel is increased at a simulation step, it is selected for upgrade by the probability distribution function.

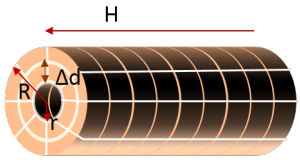


Figure 2: Basic cylindrical pipe

Fig. 1 describes our simulation and rendering pipeline. Corrosion is a generic phenomena, which broadly depends on the material properties of the object and the environmental conditions it is exposed to. The input parameters of the model are as follows:

- M : material type (steel, copper, zinc etc.)
- C : corrosivity level of the environment (1–5)
- t : elapsed time for which corrosion the process is to be simulated

We describe the simulation scheme using cylindrical pipes. Let us consider a pipe with interior radius r, exterior radius R and height H, as shown in Fig. 2. Our simulation is agnostic to the voxel size. Let the thickness of the exposed layer (voxel width) be Δd. ρ is the density of the metal and 'A' is the total area exposed ($2\Pi RH$ assuming the outer surface of the pipe is exposed). So the volume exposed is $A * \Delta d$. The weight of the exposed volume (α) is given by $A * \Delta d * \rho$. Equation 1 gives us the value of Ω, the expected weight loss (in g/m^2) in the given elapsed time. The expected weight loss in grams (ξ) is thus:

$$\xi = \Omega * A \tag{2}$$

The probability of a voxel to be removed can now be calculated as the ratio of the expected weight loss to the weight of the exposed volume. It is given by:

$$\upsilon = \frac{\xi}{\alpha} = \frac{\Omega * A}{A * \Delta d * \rho} = \frac{k * C^n * t^m}{\Delta d * \rho} \tag{3}$$

υ plays a vital role in the propagation of the corrosion phenomena. This is explained in detail in section 3.3. The material weight loss computation is agnostic to the area exposed and shape dependencies (Equation 3). The proposed model is independent of the the voxel resolution (as seen in Equation 3). The probability of removal of a voxel is also independent of the total exposed area.

3.3 Model Simulation

At every simulation step, the state of each voxel is primarily influenced by three main factors: the time of exposure, the the level of corrosion in its vicinity and the exposure of the voxel.

Corrosion rate has been measured to be an exponentially decreasing function of time [12]. We model the corrosion probability to decay with time according to Equation 4.

$$\chi = \lambda_0 * exp(-\lambda_1 * t) \tag{4}$$

with some constants λ_0 and λ_1. To achieve the expected Ω, we assign $\lambda_0 = \upsilon$ from Equation 3 to calculate χ for updating ψ. This controls the aggregate material loss in the simulation. Our simulation has fixed $\omega = 1$. The pace of corrosion is hence controlled by λ_1. This can increase or decrease the probability of material loss. [20] does not have any such control over the speed of the complete simulation.

For each exposed voxel, the average neighborhood (β) is the average of ψ values of its neighbors. Now, the corrosion simulation steps are listed below for each exposed voxel 'a':

1. Choose a random number r uniformly distributed between 0 and 1.
2. If $r < \chi$, $\psi(a)$ += δ
3. If $\beta(a) > \nu$, $\psi(a)$ += δ
4. If $\psi(a) > \omega$, remove the voxel. Distribute $\psi(a)$ equally among the now exposed voxels in its neighborhood.
5. For all exposed voxels, repeat steps 1–3 until the total material loss reaches Ω for a given elapsed time t given in Equation 3 .

If a voxel has been exposed longer, the chances of its corrosion is high. Closeness to a highly corroded region also accelerates the corrosion process of a voxel.

4 RENDERING

Corroded objects exhibit a wide spectrum of appearances, which may be nearly impossible to replicate using conventional rendering techniques. The dynamic conversion of one state to another depending on the interaction between the environment and the material further complicates rendering. To best represent the actual state of the object, we propose a general rendering framework, which imparts the final color to the corroded object derived from actual high resolution photographs of real objects. Our simulation results in the set of surface voxels whose exposure is primarily governed by material-environment interaction over a certain period of time. We estimate their appearance by that of similarly corroded points in the input photograph. We start with a parameterization of the original surface to be simulated. This allows us to map the model voxels back to that parameterization at any stage of the simulation.

 ISBN 978-80-86943-65-7

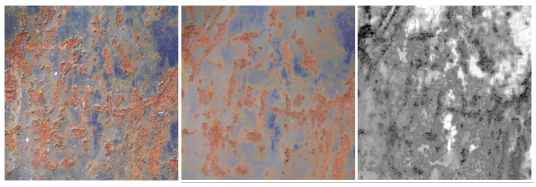


Figure 3: Real photograph sample (a), Albedo Map (b) and its corresponding weathering degree map (c)

Then, instead of re-surfacing the model voxels, we directly render the original surface with the removed voxels encoded in a displacement map, following [20]. The displacement map value is directly equal to the depth of the exposed surface from the original surface in the direction normal to that surface. The simulation also generates the current level of corrosion for every exposed voxel in a "weathering degree" map (W_2). We use this weathering degree map also, to compute a bump-map.

Finally, we compute the color by example of a corroded material of the same type. We first eliminate the illumination from the photograph [5], getting a map of albedo values for each pixel, I_1. Employing the user-in-the-loop technique of Wang et al. [24], we then construct a degree map of the photograph: the estimated corrosion level of the material at each pixel. Fig. 3 shows an example of a photograph, its albedo map and the final weathering degree map for the real photograph.

We then generate the final albedo map I_2 of the simulated object using these three maps:

1. Example weathering degree map W_1 constructed from the photograph
2. Simulated weathering degree map W_2 generated from our simulation
3. Example albedo map I_1 constructed from the photograph

Let I_1 and W_1 each be of size M $\times$ N and let I_2 and W_2 be of size P $\times$ Q.

To compute $I_2(x,y)$, one possibility is to individually locate the value (similar to) $W_2(x,y)$ in W_1 and assign the albedo from I_1 found at that pixel. This provides unnatural and random looking results. Further, it is possible to find $W_2(x,y)$ at multiple locations in W_1, with large variance in color values. Hence a coherent look-up provides a more natural look. However, instead of matching pixels based on their neighborhoods as in [24], we match entire tiles. Unfortunately, this generates discontinuity at tile boundaries. Hence, we employ overlapping tiles and then combine color from multiple tiles at each pixel. Further, we do not require a seven dimensional appearance manifold to generate realistic color. 7D input is cumbersome to acquire. We instead use

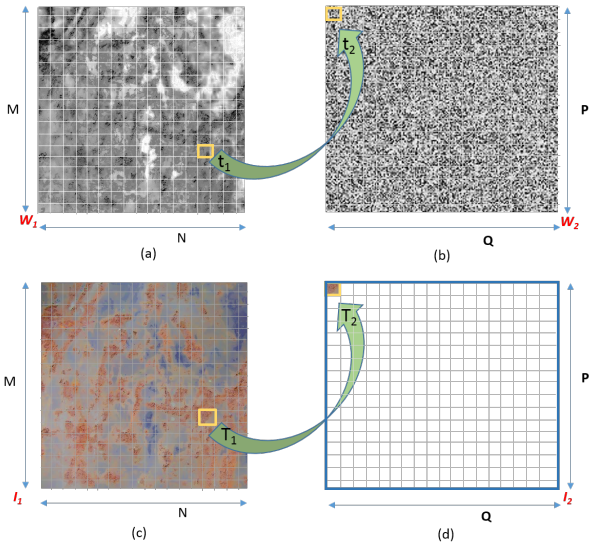


Figure 4: Procedure to generate the final diffuse color map based on least chi-squared distance between histogram distribution of degree values. Each tile (highlighted as yellow) is m*m sized.

standard RGB photographs and obtain improvements over the technique in [24] as demonstrated later. We describe our algorithm next.

We consider $m \times m$ tiles in each map. As I_1 and W_1 are of the same size, a tile T_1 at (x,y) location in I_1 corresponds to the same (x,y) location in W_1. (See Fig. 4(a) and (c), t_1 and T_1, respectively.) We hence use the same symbol to denote corresponding tiles in the two maps. Similarly, we use a common symbol for tiles in W_2 and I_2 (See Fig. 4(b) and (d), t_2 and T_2, respectively).

As mentioned, we employ a sliding window approach. We start with an $m \times m$ tile in one corner. The next tile is offset by one pixel in one of the dimensions, and so the window slides. The entire map is overlaid with such overlapping tiles (see Figure 4). For each tile in W_2, we compute the best matching tile in W_1 using its *histogram distribution of the degree values* as the 'feature vector' for matching.

- Calculate histogram of each tile in W_1 and W_2 using their degree values.
- For each tile T_2 in W_2, find the best matching tile T_1 in W_1, based on least chi-squared distance [18] between their respective histograms.
- Impart color from the matching tile T_1 in I_1 to T_2 in I_2.
- Every pixel $p \in I_2$ lies in multiple (sliding) tiles (see tiles marked in blue and green in Fig. 5) and has corresponding color contributions.
- The final color for pixel p is generated by the weighted average of these color contributions.

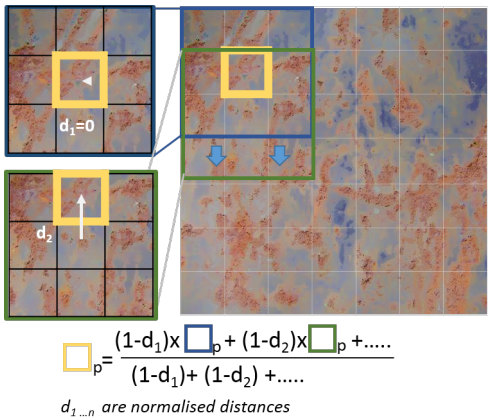


Figure 5: Sliding window procedure to impart color to a pixel p in I_2. The distribution of colors is dependent on the spatial distance of pixel p in I_2 from the center of the tile (T_2 in I_2). Yellow highlighted part is the pixel for which color contribution calculation is shown. Blue and green are the tiles pixel p is part of. For the blue tile p lies at the center, but for green tile p is at distance d_2 from the center. The color contribution from the blue tile at distance d_1 would be more than the RGB value from green tile at d_2.

The color maps generated using the RGB values employing the pixel matching technique of [24] produces blockiness. As can be observed in Fig. 6, our results produce sharper results. Moreover, our colors have a higher fidelity to the computed weather map in W_2.

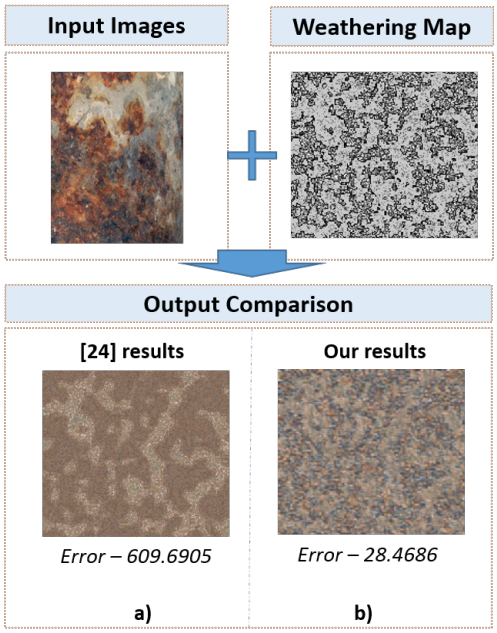


Figure 6: Comparison between our diffuse map generation and texture generated using [24], our results have much less blockiness and much better spread of colors. Also L^2 norm error is much less.

We perform an error analysis of results from our color maps and those generated by [24]. This is done by calculating how accurately a color has been assigned for a particular corrosion level. For each pixel in I_2, we compute the difference in color from the "nearest supporting color" in I_1. We then compare the L^2 norm error for both the set of outputs. To compute the nearest supporting color for $I_2(x,y)$, we locate all points on the appearance manifold [24] on I_1 and W_1, which have a corrosion level of $W_2(x,y)$. Among these, $I_1(x,y)$ that is closest (in L^2 sense) to $I_2(x,y)$ is considered the nearest supporting color. Intuitively, the corroboration of a similar color with the same corrosion level in the input maps testifies to it being the right choice.

Also, we generate a diffuse texture map, a normal map and a displacement map, which incorporate the overall deterioration in the three dimensional object, which is difficult in their proposed strategy.

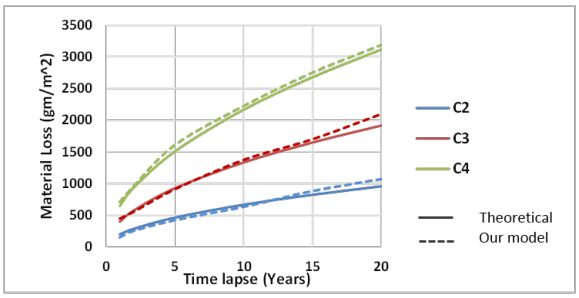
Finally, after generating the diffuse texture map, the normal map and the displacement map, we produce the final rendering.

5 RESULTS AND VALIDATION

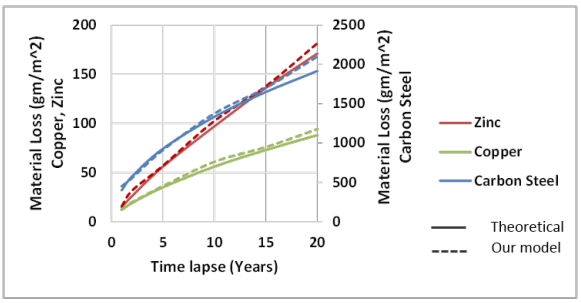
Our model results closely follow the weight loss paradigm of materials including Iron, Copper and Zinc under environmental conditions ranging from gentle to highly hostile. We validate our results exhibiting a comparison of the weight loss from our simulation to the predicted weight from the ISO standard. Our major focus is on realistic rendering of corroded objects and less on the performance statistics. Our model simulation takes 2-4 minutes (for Figs. 8, 9 and 10). Texture generation is a two step process. The first step creates the appearance manifold (which is computed just once) and the second step generates the final color, which takes 20–30 seconds for for a texture size of 256×256.

Our model simulations are agnostic to voxel size. The surface layer of depth is specified by the user. The size of voxels, in real units, is also specified by the user. For the results shown in Figs. 8, 9 and 10, the thickness of the pipe is 0.5cm. This depth is divided into sixteen layers and the average voxel size is 0.00491 cm^3. In Fig. 7(a), we compare time variant simulated results for carbon steel under various environmental conditions (C1 - C5) with predicted weight loss in grams per unit area from the ISO standard. As is apparent from the graph, our model complies with the ISO standard from various atmospheric conditions at different elapsed times. The graph clearly depicts that the total weight loss closely matches the predicted value. Being agnostic to the type of material, our framework is able predict the state of other materials like Copper and Zinc also once the respective material property is provided as input. We have validated our model with the actual weight loss in materials such as Steel, Copper and Zinc. In Fig. 7(a), we compare our results for various materials including steel, copper and zinc for cor-

 ISBN 978-80-86943-65-7



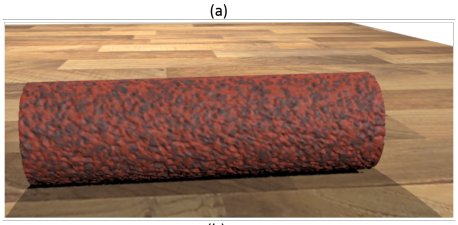
(a)



(b)

Figure 7: The two charts compare weight loss in grams per unit area from our simulation at different elapsed times with predictive weight losses from [15]. The first chart compares the weight losses for carbon steel at different corrosivity levels. (b) shows the weight losses for different metals at corrosivity level C3. The compliance of our model with standard results is evident.

(a)



(b)

Figure 8: Visual comparison of material loss in steel (a) and copper (b) pipe after one year

rosivity level C3 and observe that the predicted weight losses are achieved by our simulation effectively.

Our results are validated both with respect to the expected structural changes as well as color similarity to input. For example, Fig. 8 showcases the rendered results within the weight loss band in steel and copper pipe for the first year as suggested by ISO [15]. For steel pipe, the weight loss is much more than for copper after one year. Our model generates a time variant corrosion simulation. Fig. 9 exhibits the state of a copper pipe at different time steps impacted by an environment of corrosivity C3. The color and surface changes for the copper pipe shows the deterioration of the pipe at multiple time steps. Fig. 10 shows the impact of different corrosivity levels on steel pipe. C5 is much more aggressive and hostile environment and the surface deterioration of the pipe clearly depicts that.

6 CONCLUSION

We propose a physio-chemically based stochastic model to predict shape degradation of objects undergoing corrosion. Our proposed approach is generic across different materials and handles different object sizes and shape intricacies. We allow for material type and environmental conditions to be given as inputs making the framework simple yet comprehensive. In order to generate rendered output we propose a holistic rendering approach involving mapping of surface with color seen in actual corroded objects. We have presented a model that estimates structural damage due to material loss for uniform corrosion of various materials. Extending the model to other forms of corrosion including pitting and crevice corrosion would be an important avenue for further research. Our current rendering scheme focuses primarily on surfaces of revolution. We aim to generate simulation for general objects in future. Also, further work can be done to understand the physical and optical properties of corrosion residue.

7 REFERENCES

[1] Bandeira, D., Walter, M.: Synthesis and transfer of time-variant material appearance on images. In: Computer Graphics and Image Processing (SIBGRAPI), 2009 XXII Brazilian Symposium on, pp. 32–39. IEEE (2009)

[2] Chen, Y., Cohen, J., Kumar, S.: Visualization of time-varying curvilinear grids using a 3D warp texture. In: Vision, Modeling, and Visualization (2005)

[3] Chen, Y., Xia, L., tsin Wong, T., Tong, X., Bao, H., Guo, B., yeung Shum, H.: Visual simulation of weathering by γ-ton tracing. In: ACM SIGGRAPH, pp. 1127–1133 (2005)

[4] Clément, O., Benoit, J., Paquette, E.: Efficient editing of aged object textures. In: Proceedings of the 5th international conference on Computer graphics, virtual reality, visualisation and interaction in Africa, pp. 151–158. ACM (2007)

[5] Dong, Y., Tong, X., Pellacini, F., Guo, B.: AppGen: Interactive material modeling from a single image. ACM Trans. Graph. **30**(6), 146:1–146:10 (2011). DOI 10.1145/2070781.2024180

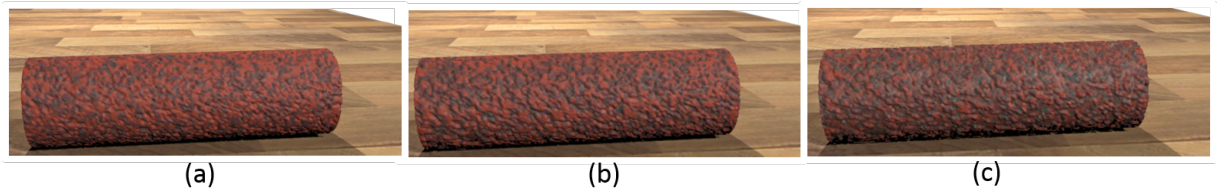


Figure 9: Depiction of time variant decay of a copper pipe at 1 (a), 2 (b) and 5 (c) years respectively. The surface degradation of the copper pipe in the 5^{th} year compared to the 1^{st} year is shown.

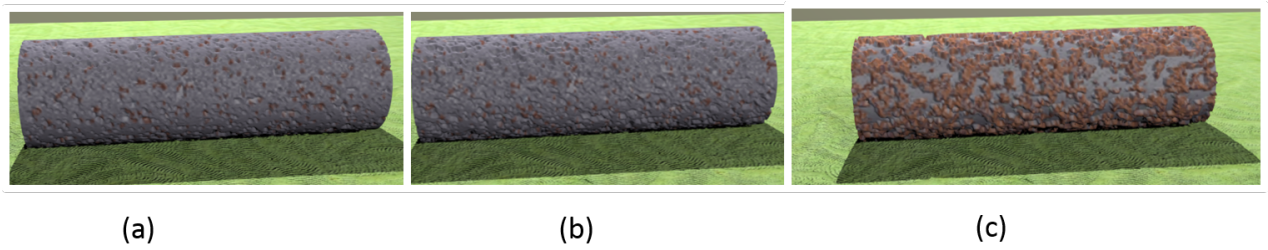


Figure 10: Corroded steel pipe after one year under C2 (a), C3 (b) and C5 (c) corrosivity levels, respectively. The pipe decays at a higher pace under C5 atmospheric conditions.

[6] Dorsey, J., Edelman, A., Jensen, H.W., Legakis, J., Pedersen, H.K.: Modeling and rendering of weathered stone. In: ACM SIGGRAPH 2006 Courses, p. 4. ACM (2006)

[7] Dorsey, J., Hanrahan, P.: Modeling and rendering of metallic patinas. In: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, pp. 387–396. ACM (1996)

[8] Dorsey, J., Pedersen, H.K., Hanrahan, P.: Flow and changes in appearance. In: ACM SIGGRAPH 2005 Courses, p. 3. ACM (2005)

[9] El Aal, E.A., El Wanees, S.A., Diab, A., El Haleem, S.A.: Environmental factors affecting the corrosion behavior of reinforcing steel III. measurement of pitting corrosion currents of steel in Ca $(OH)_2$ solutions under natural corrosion conditions. Corrosion Science **51**(8), 1611–1618 (2009)

[10] Genel, K., Demirkol, M., Gülmez, T.: Corrosion fatigue behaviour of ion nitrided aisi 4140 steel. Materials Science and Engineering: A **288**(1), 91–100 (2000)

[11] Guessasma, S., Elkedim, O., Nardin, P., Hamzaoui, R., Grosdidier, T.: Monte carlo simulation of uniform corrosion process under potentiostatic conditions. Corrosion science **49**(7), 2880–2904 (2007)

[12] Henshall, G., Halsey, W., Clarke, W., McCright, R.: Modeling pitting corrosion damage of high-level radioactive-waste containers, with emphasis on the stochastic approach. Tech. rep., Lawrence Livermore National Lab., CA (United States) (1993)

[13] Hwang, G., Yoon, S.H., Park, S.: Video-based weathering gallery. Multimedia Tools and Applications pp. 1–17 (2015)

[14] Corrosion of metals and alloys – corrosivity of atmospheres – classification, determination and estimation (2012)

[15] Corrosion of metals and alloys – corrosivity of atmospheres – guiding values for the corrosivity categories (2012)

[16] Kadry, S.: Corrosion analysis of stainless steel. Eur. J. Sci. Res **22**(4), 508–516 (2008)

[17] Kamata, Y., Manabe, Y., Yata, N.: Simulation of aging metal with preservative coating. In: The 2013 International Conference on Computer Graphics, Visualization, Computer Vision, and Game Technology. Atlantis Press (2013)

[18] McCune, B., Grace, L., Urban, D.: Analysis of Ecological Communities. MjM Software Design, Gleneden Beach, OR (2002)

[19] Mérillou, S., Dischler, J.M., Ghazanfarpour, D.: Corrosion: simulating and rendering. In: Graphics Interface, vol. 2001, pp. 167–174 (2001)

[20] Nisha, J., Prem, K., Kumar, S.: Simulation and rendering of pitting corrosion. In: Computer Vision, Graphics & Image Processing, 2014. ICVGIP'14 (2014)

[21] Rivas, D., Caleyo, F., Valor, A., Hallen, J.: Extreme value analysis applied to pitting corrosion experiments in low carbon steel: Comparison of block maxima and peak over threshold approaches. Corrosion Science **50**(11), 3193–3204 (2008)

[22] Scarf, P.A., Laycock, P.J.: Estimation of extremes in corrosion engineering. Journal of applied statistics **23**(6), 621–644 (1996)

[23] Shreir, L.L., et al.: Corrosion. Vol. I. Metal/environment reactions. Ed. 2. Butterworth & Co.(Publishers) Ltd. (1976)

[24] Wang, J., Tong, X., Lin, S., Pan, M., Wang, C., Bao, H., Guo, B., Shum, H.Y.: Appearance manifolds for modeling time-variant appearance of materials. ACM Transactions on Graphics (TOG) **25**(3), 754–761 (2006)

[25] Wojtan, C., Carlson, M., Mucha, P.J., Turk, G.: Animating corrosion and erosion. In: NPH, pp. 15–22. Citeseer (2007)

[26] Xuey, S., Wang, J., Tong, X., Dai, Q., Guo, B.: Image-based material weathering. Computer Graphics Forum **27**(2), 617–626 (2008)