

노이즈와 오실레이터

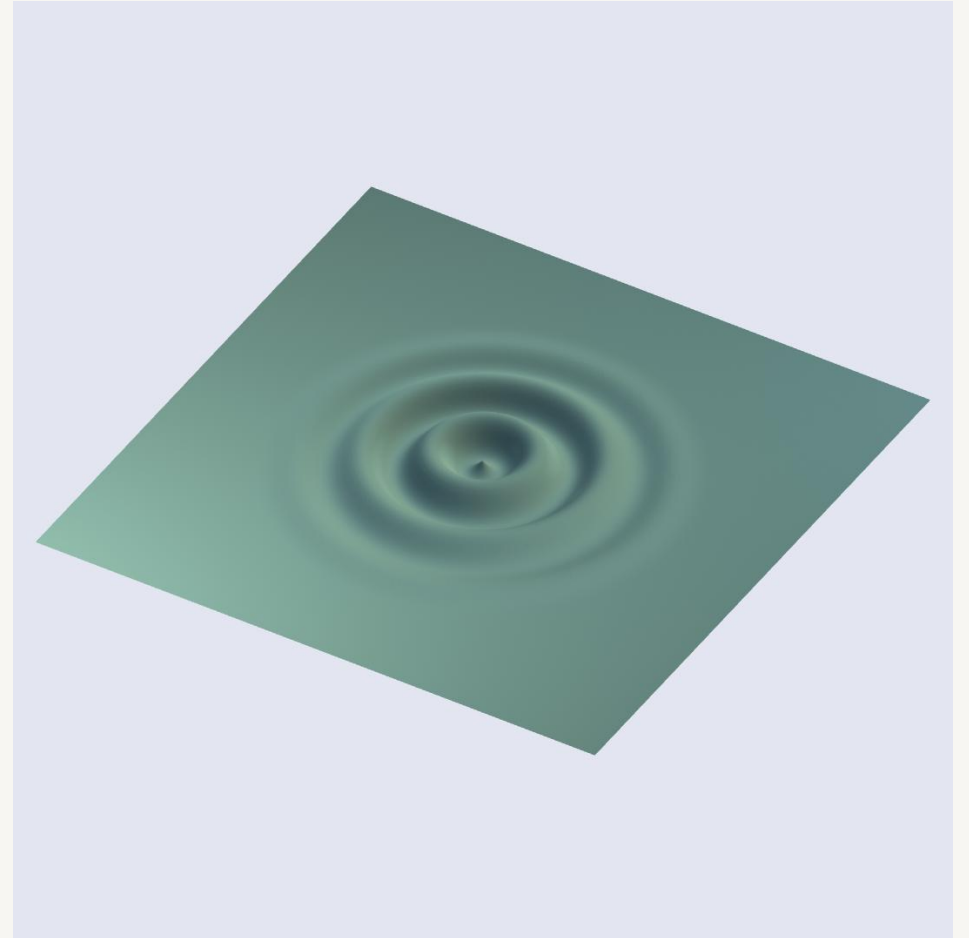
숫자를 이미지와 소리로 읽고, 원형 물결로 연결하기

소리로 움직이는 원형 물결

Audio Oscillator · Math · RMS

Ramp TOP · Height Map · Phong MAT

단국대학교 · 120분



오늘의 수업 흐름

기본 개념을 익힌 뒤, 하나의 예제로 끝까지 연결합니다

00-20분 값·이미지·신호·높이·표면 방향

20-55분 소리 크기 분석과 원형 무늬 만들기

55-65분 휴식과 중간 저장

65-100분 높이 변형·재질·조명·렌더

100-115분 값 변경 비교와 짧은 공유

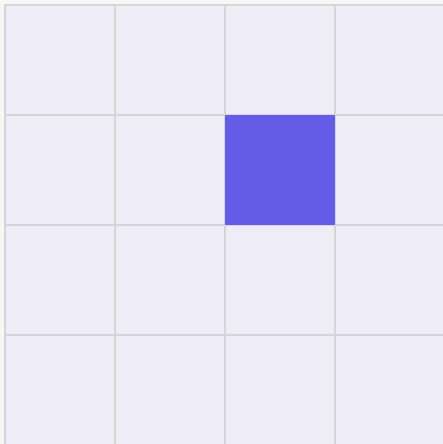
115-120분 저장과 제출

필수 수업: 1-34장 / 35장 이후: 설정 참고와 선택 심화

좌표의 값이 픽셀의 밝기가 됩니다

픽셀마다 계산하고, 모든 픽셀을 모으면 한 장의 이미지가 됩니다.

픽셀의 위치



(x, y)

좌표에서 읽은 값

$n = 0.7$

RGB로 저장



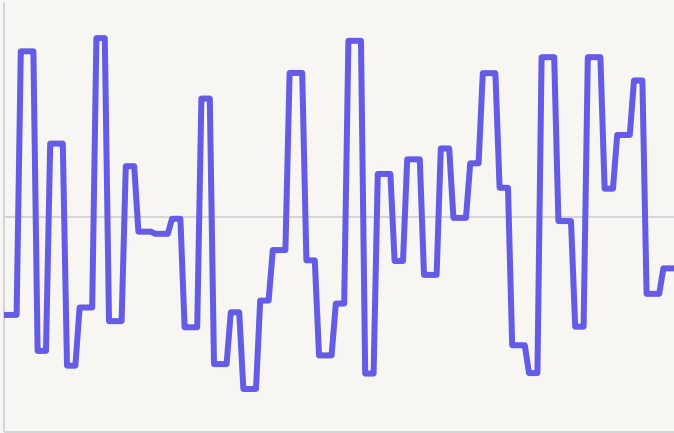
$(0.7, 0.7, 0.7)$

예측 RGB의 세 값을 모두 0.2로 낮추면 어떻게 보일까요?

무작위·노이즈·주기적 신호

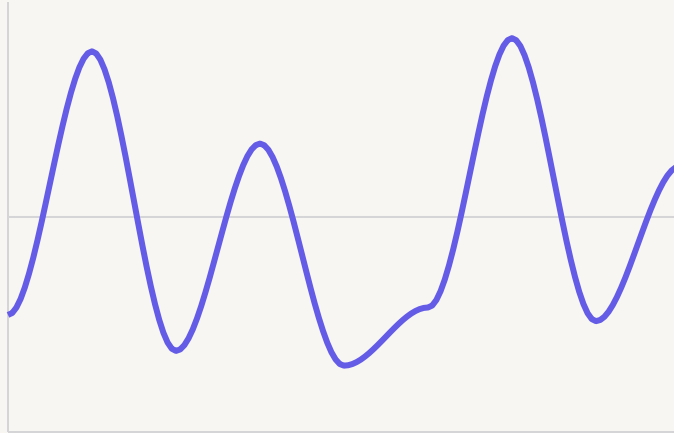
옆의 값과 어떤 관계가 있는지, 같은 모양이 되풀이되는지 살펴봅니다.

독립적인 무작위



이웃 값의 관계가 약함

연속적인 노이즈



가까운 입력에서 비슷한 값

주기적인 진동



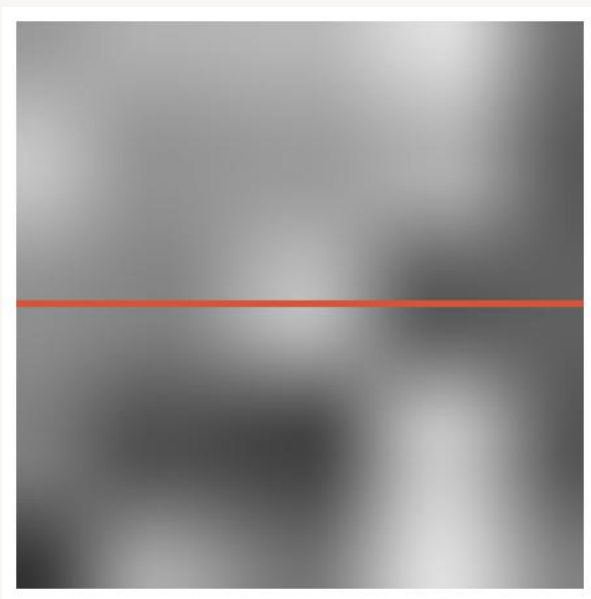
일정 간격으로 반복

예측 구름의 윤곽과 규칙적인 깜빡임에는 각각 무엇을 고를까요?

Noise TOP과 Noise CHOP

이미지의 픽셀 배열과, 이름이 있는 숫자 채널을 구분합니다.

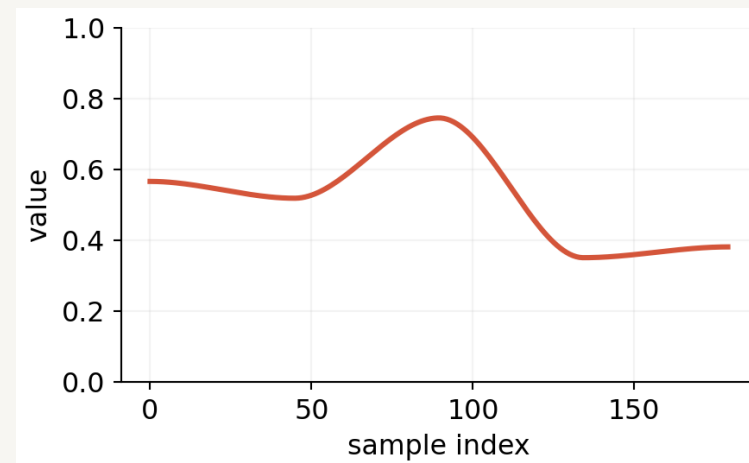
TOP · 위치별 RGBA



TOP to CHOP

선택한 픽셀을 숫자로

CHOP · 채널별 샘플

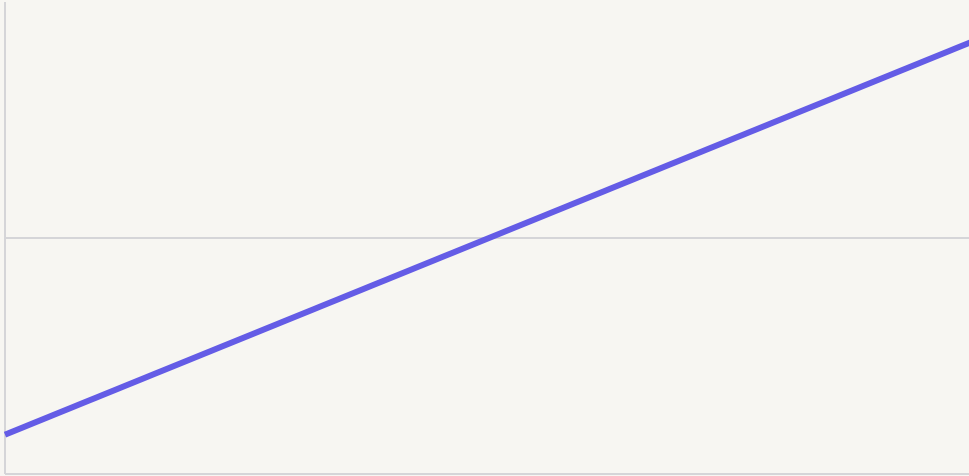


Noise CHOP을 새로 만드는 것과, TOP의 픽셀을 읽어오는 것은 다릅니다.

시간을 반복되는 값으로 바꾸기

오실레이터는 반복되는 값을 만들어 움직임이나 소리를 만드는 장치입니다.

일정한 증가



시간이 갈수록 계속 커짐
`absTime.seconds`

반복되는 변화

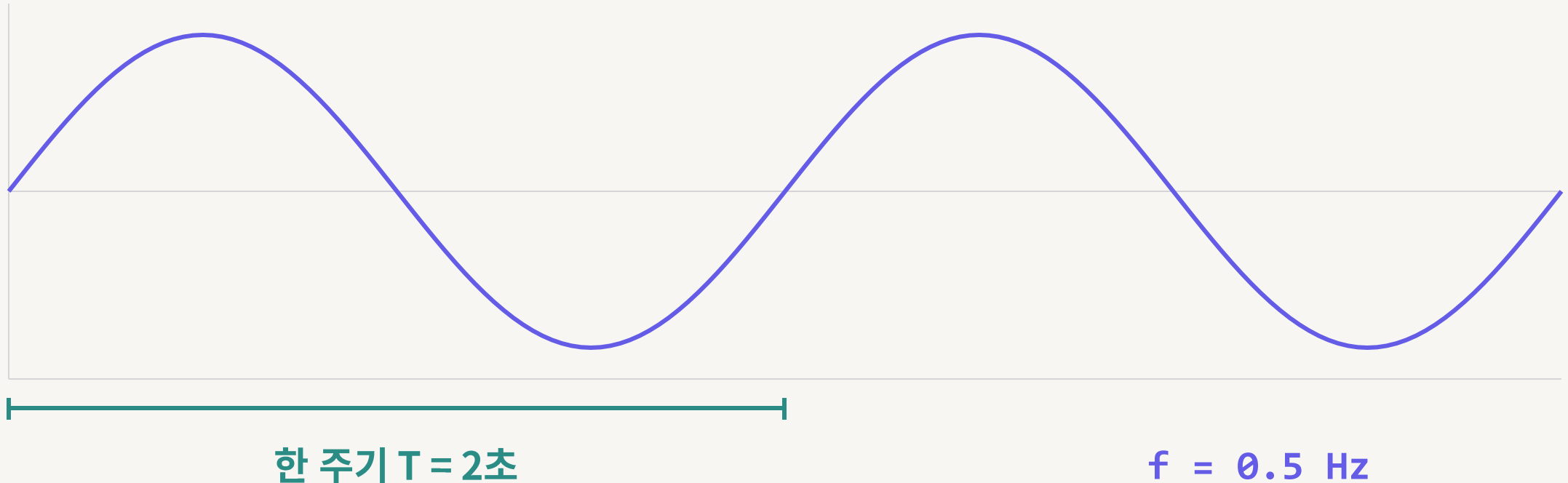


최솟값과 최댓값 사이를 반복
오실레이터

예측 원의 반지름에 연결했을 때, 화면 밖으로 계속 커지는 쪽은 무엇일까요?

주기 T와 주파수 f

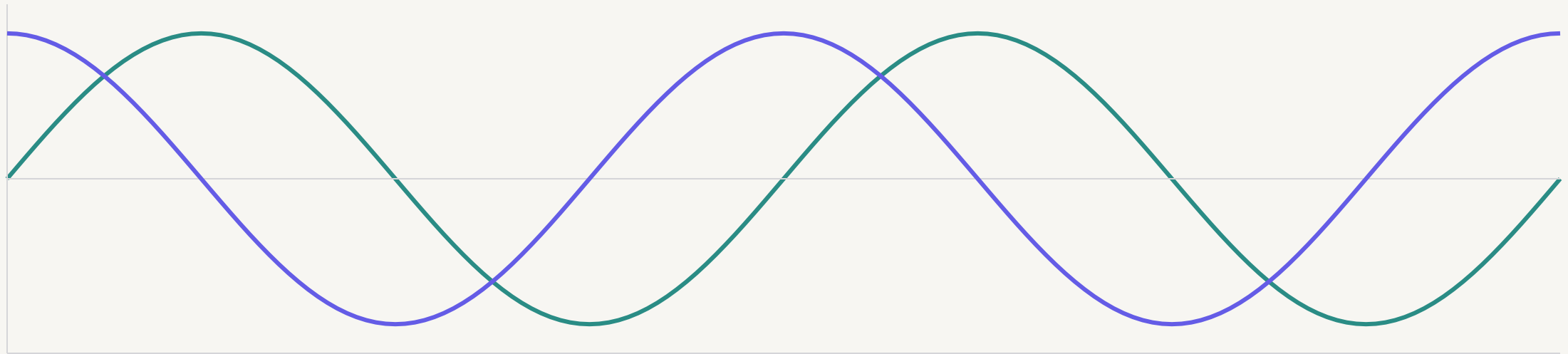
주기는 한 번 반복하는 시간, 주파수는 1초 동안의 반복 횟수입니다.



예측 0.5 Hz를 1 Hz로 바꾸면 한 번 커졌다 작아지는 시간은 얼마일까요?

위상 · 반복 안에서의 위치

같은 속도와 같은 범위에서도, 시작하는 지점을 다르게 할 수 있습니다.



청록: 위상 0

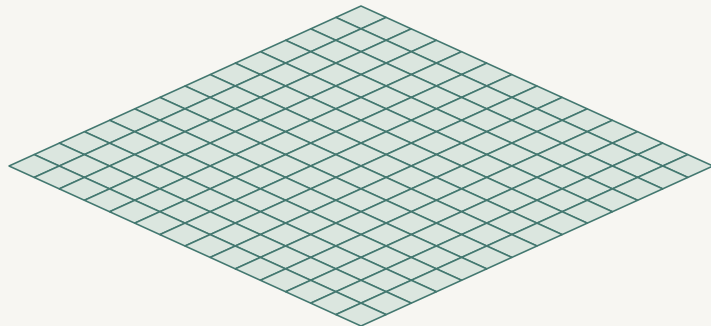
보라: 위상 0.25주기

예측 두 원이 번갈아 커지게 하려면 어느 정도의 위상 차이가 좋을까요?

같은 평면에서 높이만 바꾸기

x와 y는 그대로 두고, 노이즈값 n 을 z축 높이에 적용합니다.

변형 전



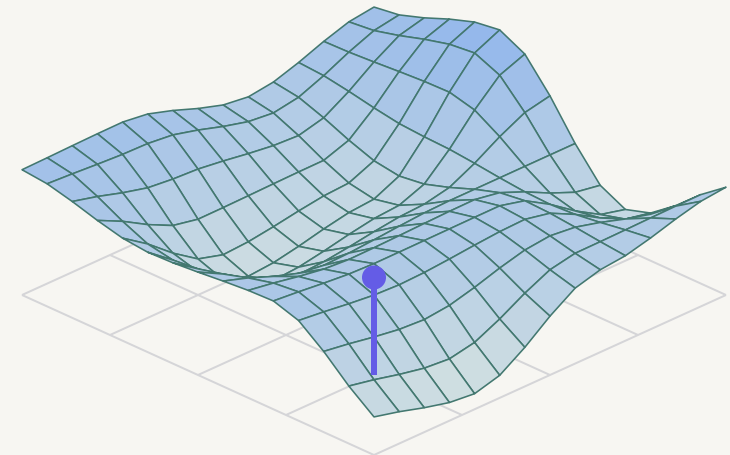
$$z = 0$$

높이로 매핑



$$\times 0.8$$

변형 후



$$z = 0.8 \times n$$



$$n = 0 \rightarrow z = 0$$



$$n = 0.5 \rightarrow z = 0.4$$



$$n = 1 \rightarrow z = 0.8$$

예측 0.8을 0.2로 줄이면, 무늬의 위치와 높이는 각각 어떻게 바뀔까요?

Normal은 표면에 수직인 방향입니다

빛을 향한 방향이 달라지면, 같은 재질에서도 밝기와 반사가 달라집니다.



예측 표면의 높이가 같아도 기울기가 다르면 법선이 같을까요?

초보용 물결 실습 시작

준비된 파일을 열고, 노드의 역할과 값의 변화를 확인합니다

W03_초보용_노드펼침.toe

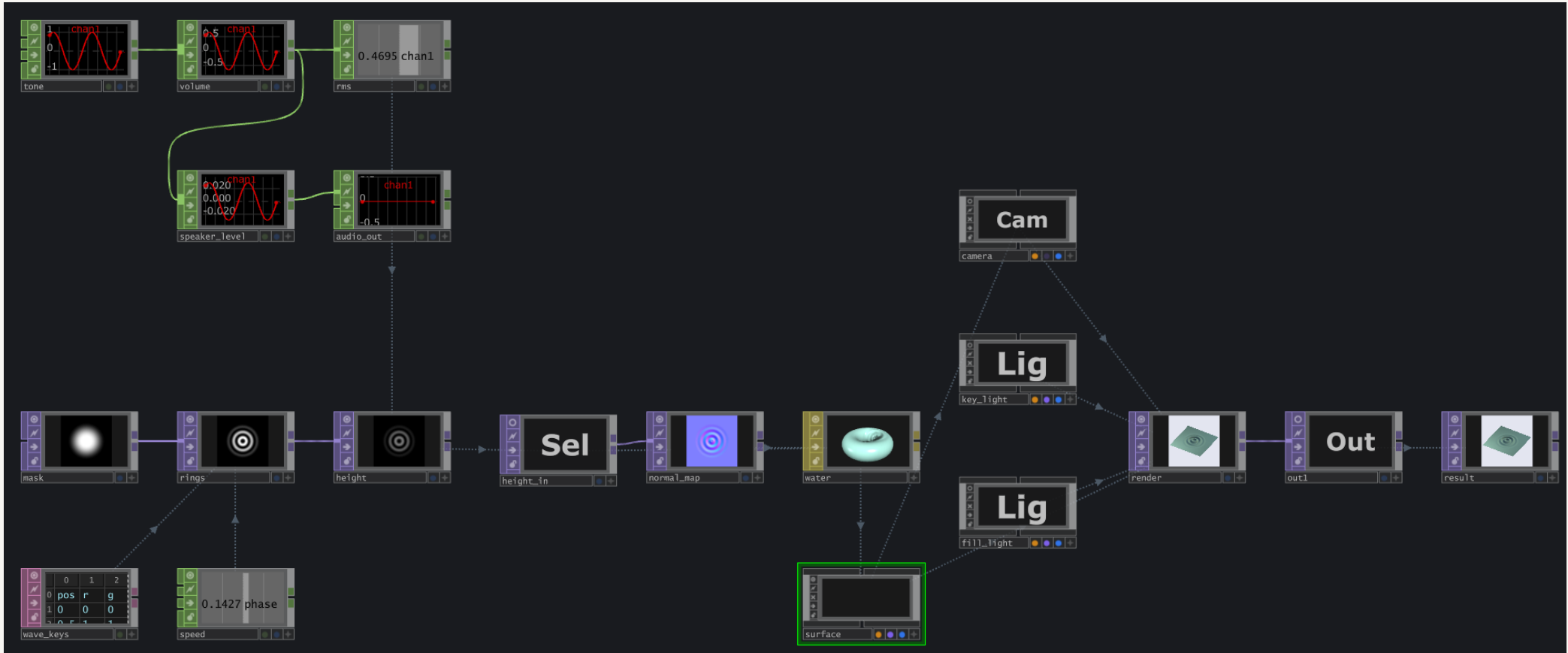
네트워크: /W03_BEGINNER_WATER

- ① tone · volume · rms에서 소리 크기 읽기
- ② mask · rings · speed에서 반복 무늬 만들기
- ③ height · water · surface로 높이 적용하기
- ④ render · result에서 화면 확인하기

설정표 36-45장 · 숫자를 복사하기 전에 바꿀 결과를 먼저 예측합니다.

전체 네트워크 읽기

같은 화면에서 소리·이미지·재질·렌더의 위치를 찾습니다



위쪽: 소리와 크기 아래쪽: 물결 이미지 오른쪽: 재질과 렌더

와이어 연결과 파라미터 참조

데이터를 선으로 전달하는 경우와, 다른 노드의 값을 읽는 경우

tone → volume → rms

CHOP 와이어

mask → rings → height

TOP 와이어

rms → height의 Brightness 1

speed → rings의 Phase

tone · 소리의 반복 신호

Audio Oscillator CHOP · 사인파가 만드는 오디오



Type = Sine

Base Frequency = 110 Hz

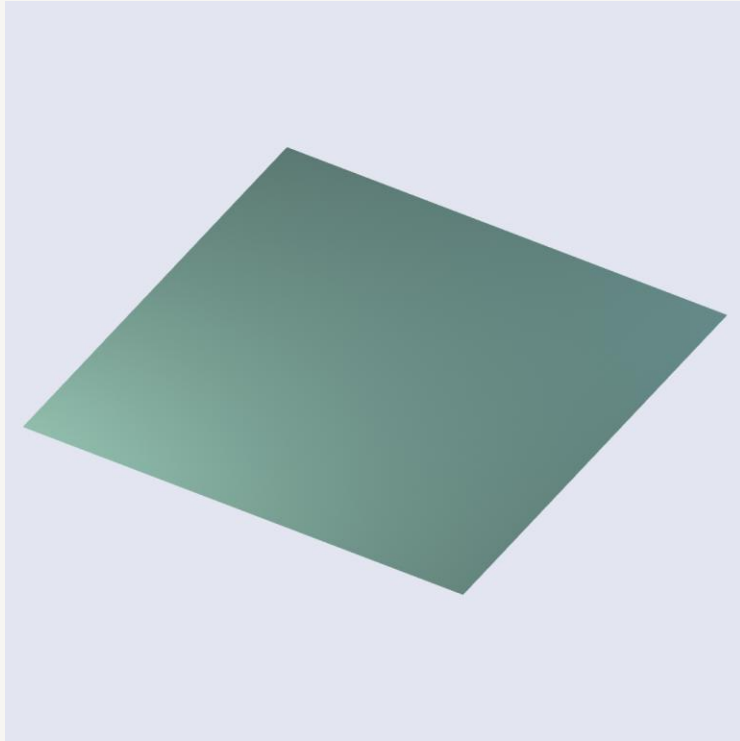
Amplitude = 1

Sample Rate = 44100

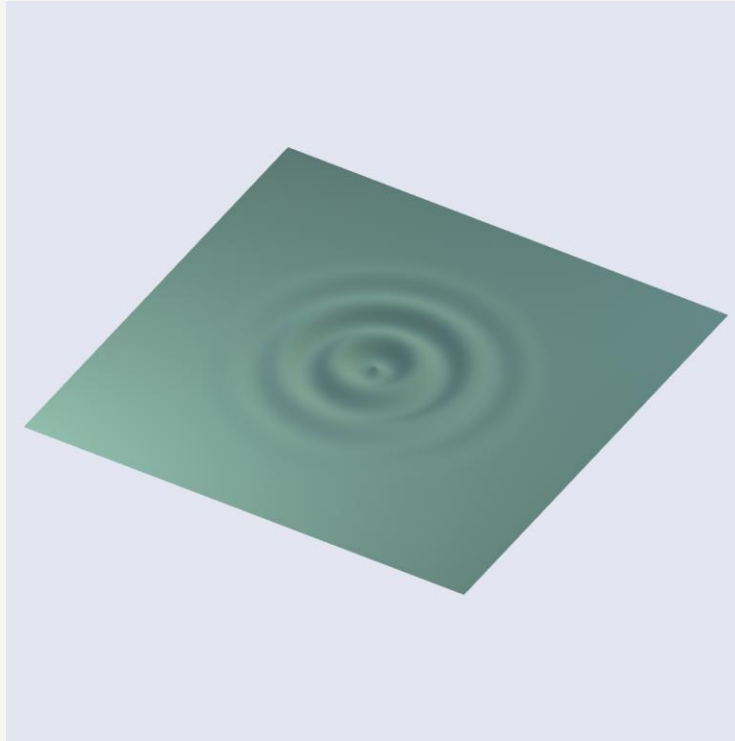
110 Hz는 소리의 음높이입니다. 화면의 속도는 speed가 정합니다.

volume · 소리와 물결의 크기

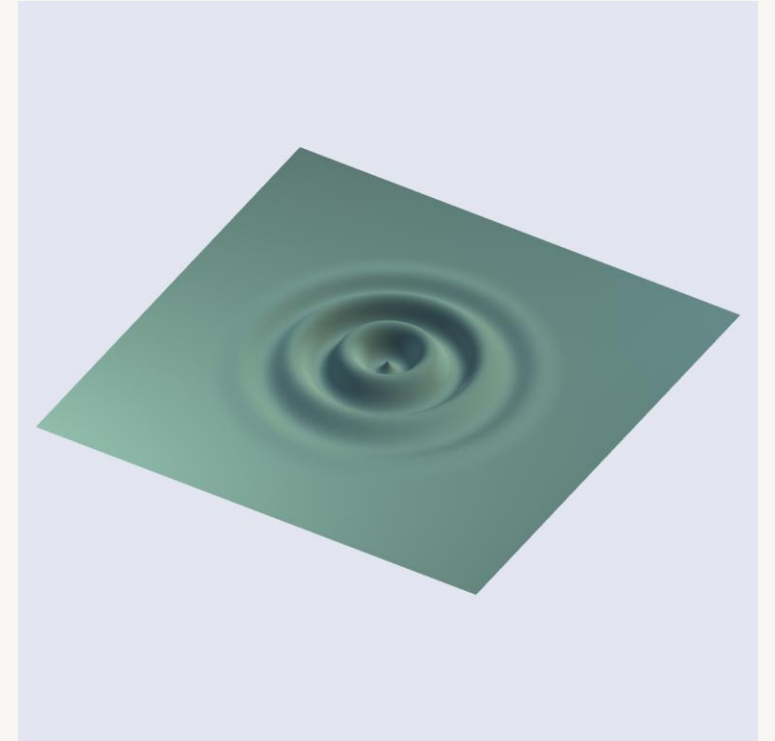
Math CHOP의 Multiply만 변경 · 위상과 카메라 고정



Multiply = 0



Multiply = 0.3

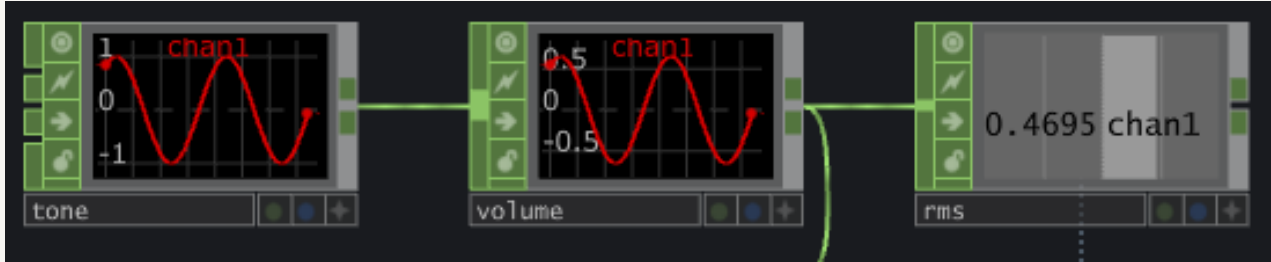


Multiply = 0.8

예측: Multiply를 두 배로 바꾸면 height 이미지의 밝기는 어떻게 될까?

rms · 오디오 신호의 크기를 숫자로 읽기

Analyze CHOP · Function = RMS Power



입력: 빠르게 오르내리는 오디오

출력: 그 구간의 크기를 나타내는 값

이 값을 물결 높이에 연결합니다.

volume 0.65 → rms 약 0.46 → height 밝기

스피커가 꺼져 있어도 신호가 들어오면 rms 값은 계산됩니다.

스피커 출력과 화면 제어

speaker_level은 스피커로 가는 신호에만 적용됩니다



speaker_level · Math CHOP

Multiply = 0.045

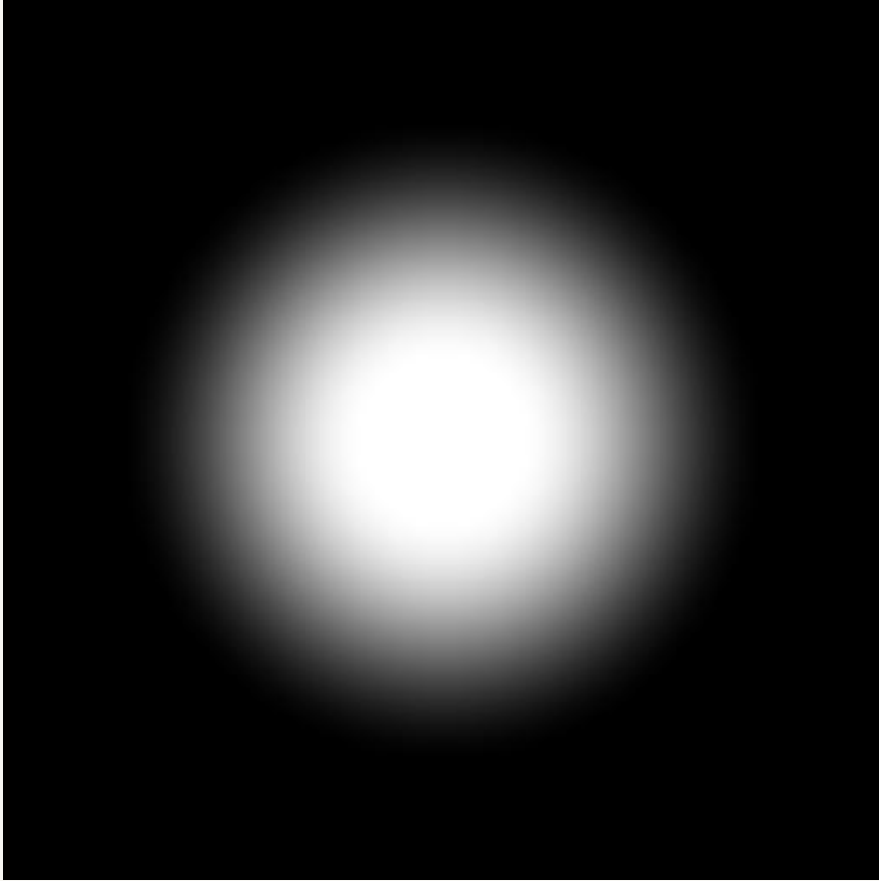
audio_out · Active On

출력 장치 선택

speaker_level을 바꿔도 물결 높이는 변하지 않습니다.

mask · 물결이 나타나는 영역

Circle TOP · 흰색은 통과, 검정은 0으로 제한



Radius X / Y = 0.37

Softness = 0.28

Softness Unit = Fraction

해상도 = 512×512

왜 경계를 부드럽게 할까? 물결이 가장자리에서 서서히 낮아집니다.

rings · 동심원 높이 무늬

Ramp TOP에서 검정-흰색-검정을 반복합니다



wave_keys: 0 → 검정 / 0.5 → 흰색 / 1 → 검정

Type = Circular

Period = 0.22

Extend Left / Right = Repeat

Interpolate = Ease-in, Ease-out

Output 페이지

Combine with Input = Composite and Set Resolution

Operation = Multiply

Extend · 패턴 범위 밖을 채우는 방법

Ramp TOP 실제 출력 · 원래 구간은 화면 가운데 0.33-0.67

Hold

시작과 끝 값을 계속 유지



Repeat

같은 방향으로 반복



Mirror

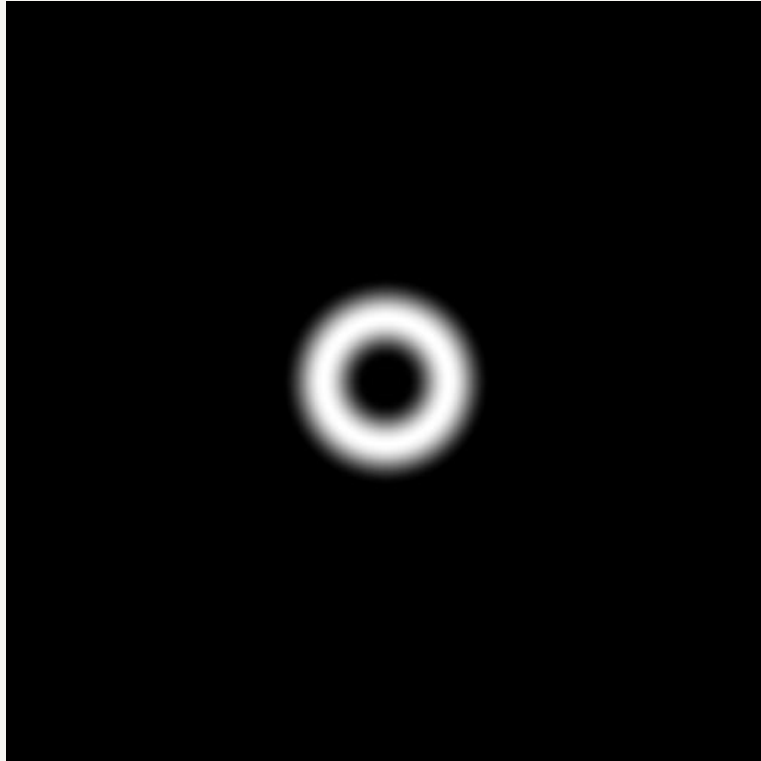
다음 구간을 뒤집어서 반복



고정: Horizontal / Period 0.34 / Phase 0.33 / Linear / 회색 0.2 → 흰색 1

물결 예제에서 Repeat를 쓰는 이유

실제 rings 출력 · Phase 0.25와 나머지 조건 고정



Hold · 한 구간 뒤에는 검정 유지



Repeat · 검정-흰색-검정 반복

현재 패턴은 좌우 대칭이어서 Mirror와 Repeat의 차이가 잘 안 보일 수 있습니다.

speed · 위상을 움직이는 신호

LFO CHOP · Ramp형 신호를 rings의 Phase에 연결



Phase = 0.1



Phase = 0.4

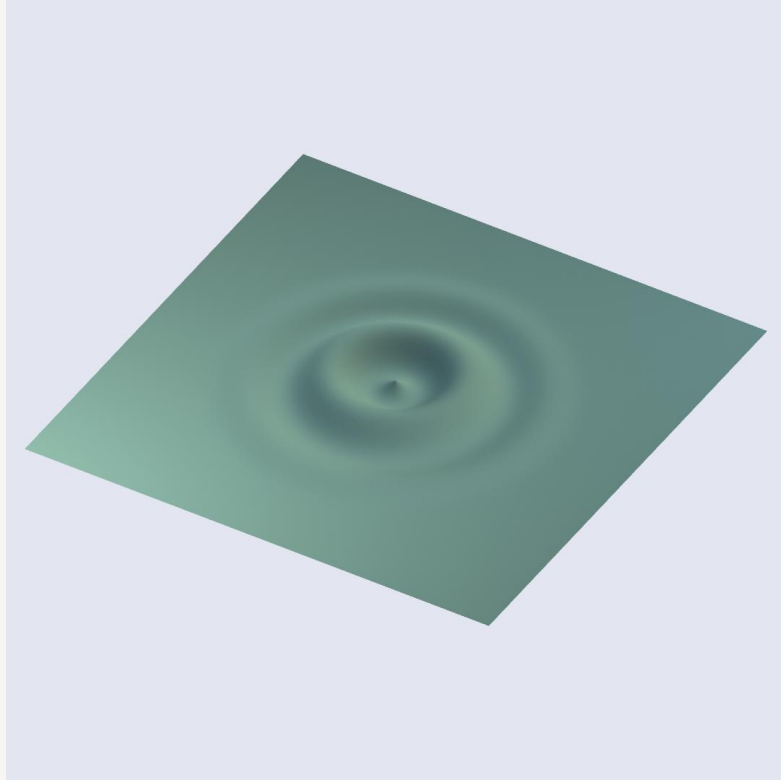


Phase = 0.7

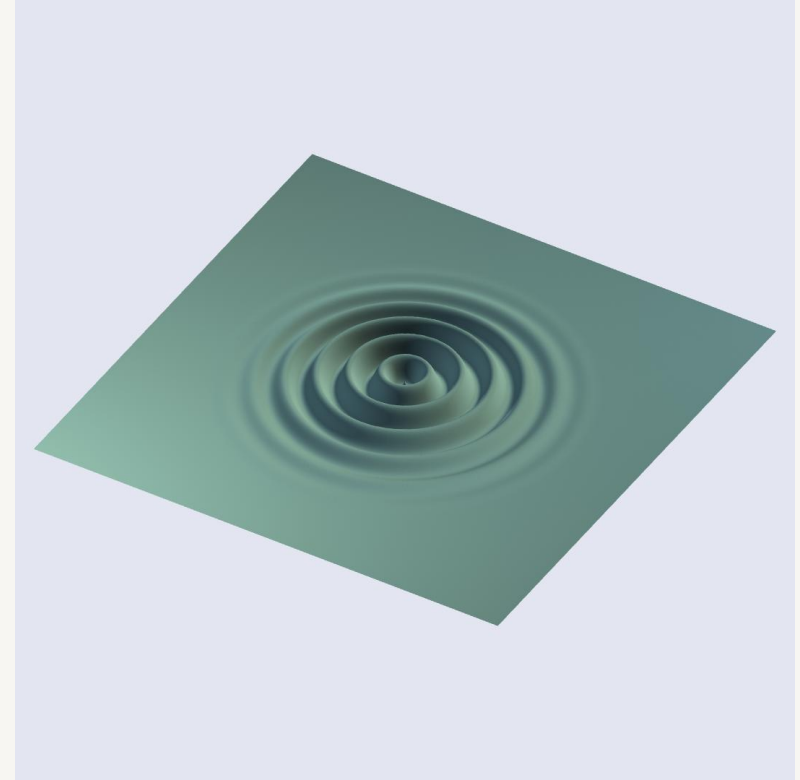
speed: Type = Ramp / Frequency = 0.22 / Channel Name = phase

rings Period · 물결 사이의 간격

같은 음량·위상·카메라에서 Period만 비교



Period 0.35 · 넓은 간격



Period 0.12 · 좁은 간격

Frequency와 Period 중, 화면을 촘촘하게 만드는 값은 어느 것일까?

휴식과 중간 저장

55-65분 · 이어서 높이와 재질에 연결합니다

10분 휴식

W03_이름.toe로 중간 저장

막혔다면 정본 파일을 다시 열어 합류

재개할 때 확인: rings가 움직이고 rms 값이 보이는가?

height · 소리 크기를 밝기로 읽기

Level TOP · Brightness 1에 rms의 chan1을 연결



volume 0.3



volume 0.8

같은 무늬

다른 밝기

밝은 값 → 더 큰 높이

Height Map과 Normal Map

높이값과 표면 방향은 서로 다른 정보를 담습니다



height_in · 높이 데이터

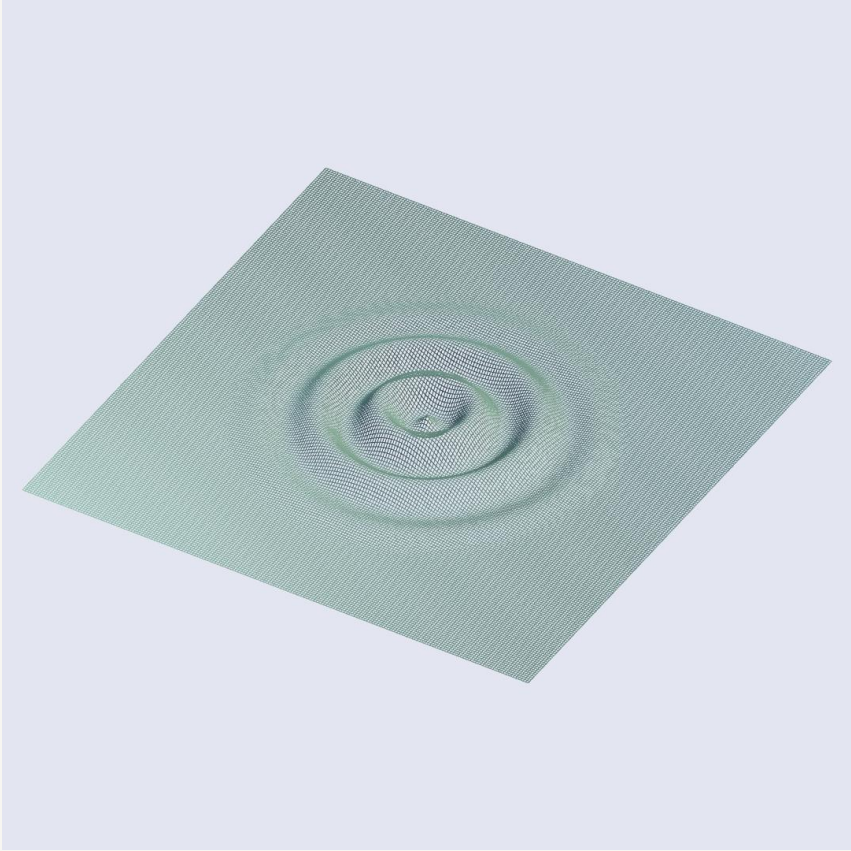


normal_map · 기울기 데이터

Slope TOP: Red = 가로 변화 / Green = 세로 변화 / Blue = One

surface · 촘촘한 평면과 표면 방향

Geometry COMP 내부: grid_mesh → surface_tangents



Grid POP · 201×201

Size = 3.4×3.4

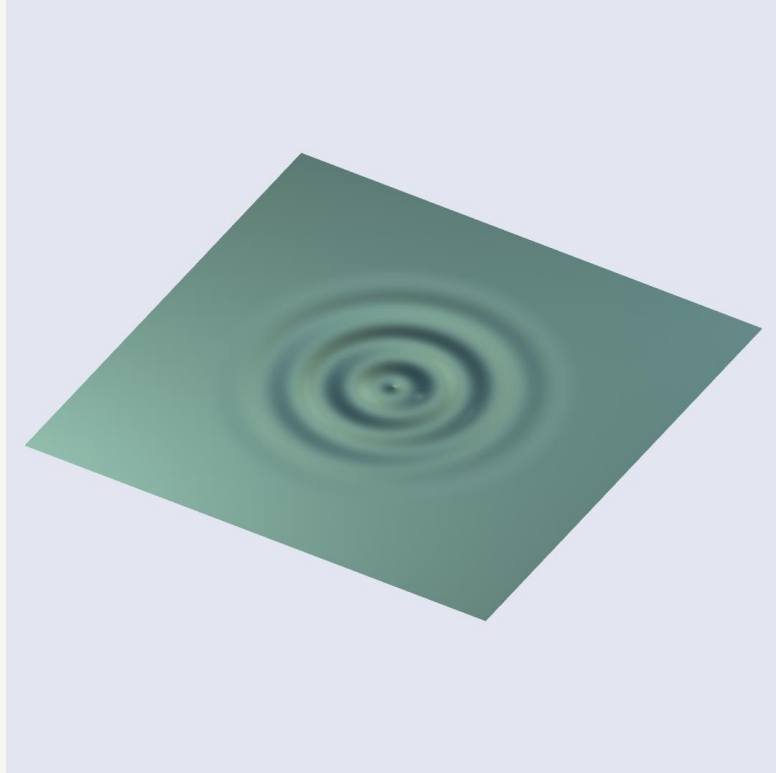
Rotate X = -90°

Attribute POP · $T = (1, 0, 0, 1)$

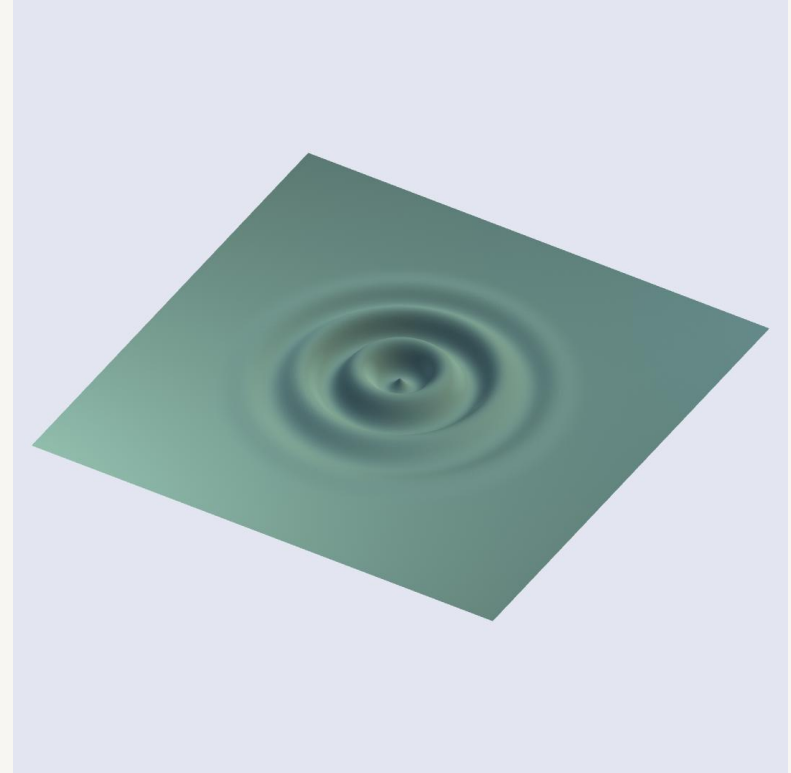
surface의 Material에 water를 지정합니다.

water · 이미지를 실제 높이로 적용

Phong MAT · Enable Height Map + Displace Vertices



Displace Scale = 0

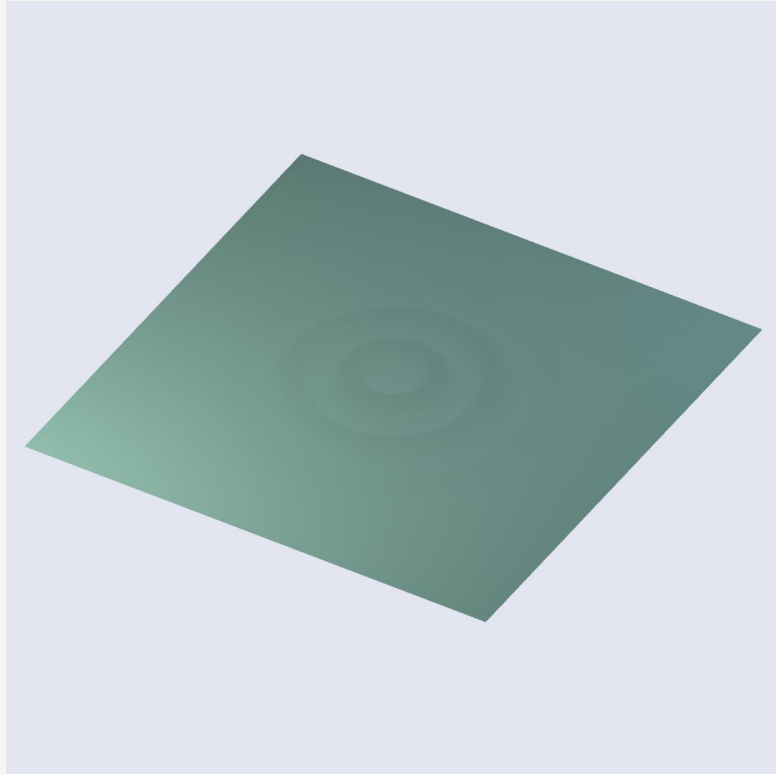


Displace Scale = 0.35

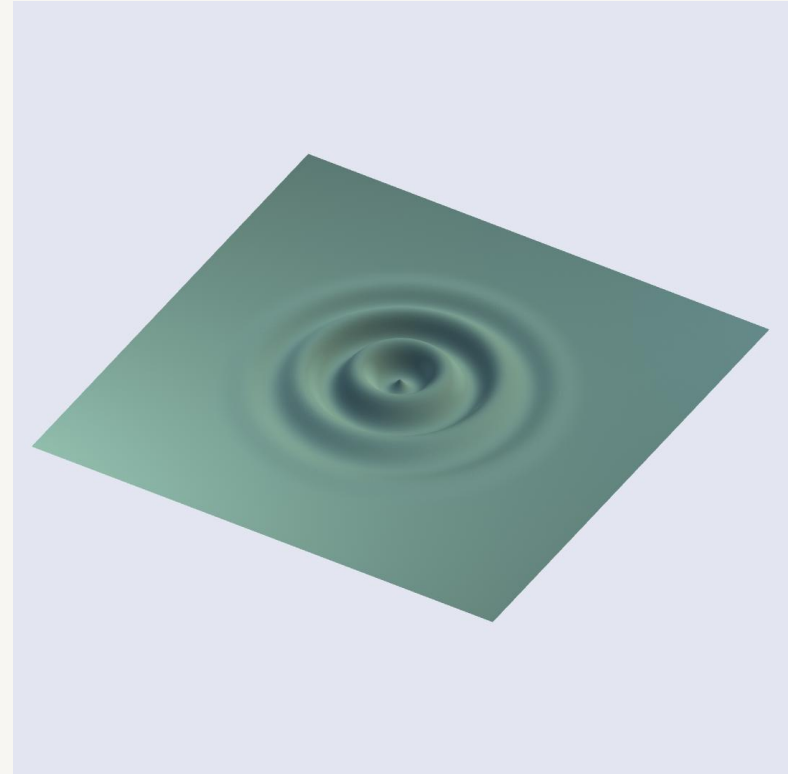
Height Map = height_in / Displace Midpoint = 0 / Parallax Scale = 0

normal_map · 기울기에 따른 명암

같은 높이에서 Slope Strength만 바꿉니다



Strength 0 · 평평한 방향 정보

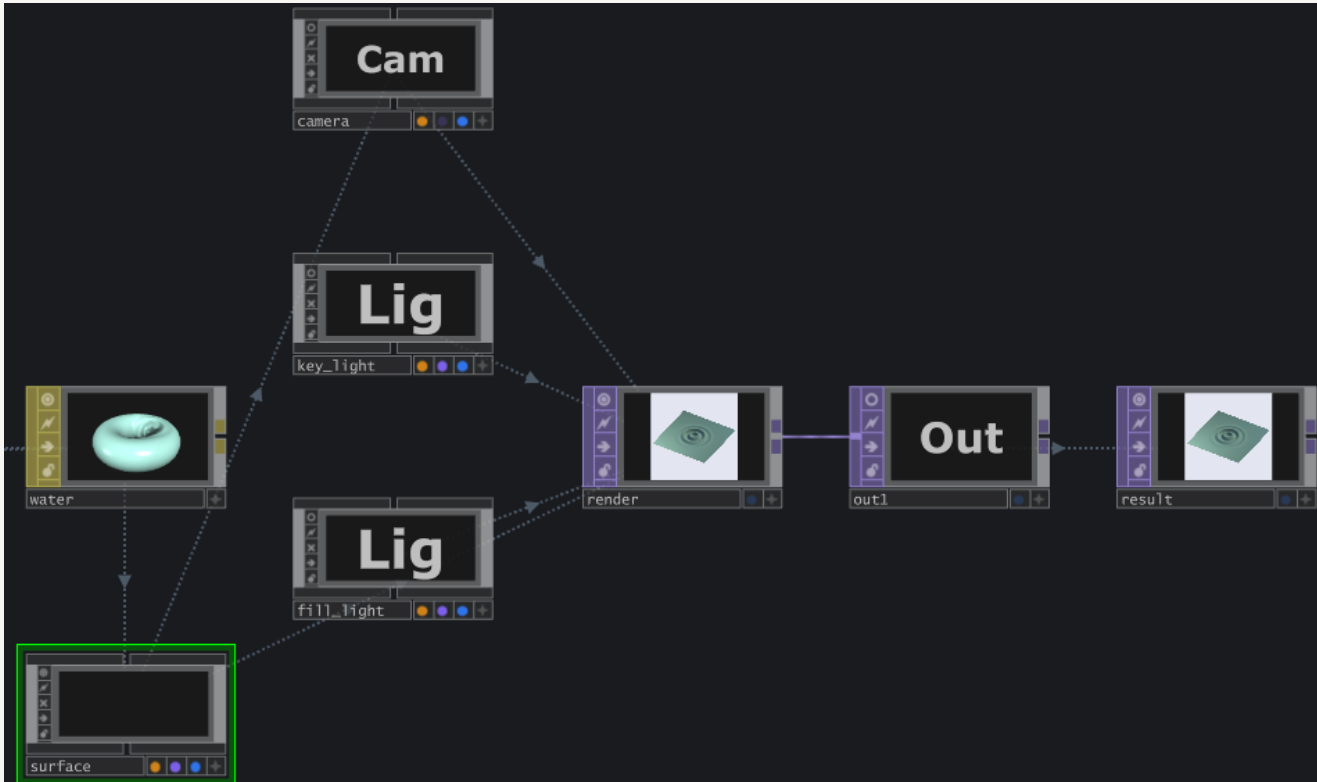


Strength 28 · 굴곡을 읽는 명암

Strength는 법선맵의 기울기 표현입니다. 물결 높이는 Displace Scale로 바꿉니다.

카메라·조명·렌더의 연결

render가 surface, camera, 두 light를 읽습니다



Geometry = surface

Camera = camera

Lights = key_light fill_light

Resolution = 1080 × 1080

render → out1 → result : result를 열어 최종 화면을 확인합니다.

완성 체크포인트

세 지점에서 확인하면 오류 위치를 좁힐 수 있습니다

- ① 소리 volume을 바꾸면 rms 숫자가 바뀌는가?
- ② 이미지 rms가 커지면 height가 밝아지는가?
- ③ 입체 volume=0이면 result가 평면이 되는가?

기본값 복구

volume 0.65 speed 0.22 rings Period 0.22

확인 문제 · 어떤 값을 바꿀까?

결과를 먼저 떠올리고, 바꿀 노드와 파라미터를 고릅니다

1. 높이는 유지하고 물결만 더 빠르게 움직이고 싶다.
2. 소리 크기는 유지하고 물결을 더 촘촘하게 만들고 싶다.
3. 화면은 유지하고 스피커 소리만 줄이고 싶다.

각자 노드명과 파라미터를 적고, 옆 사람에게 이유를 설명합니다.

풀이 · 값의 역할과 연결 위치

어디에 연결했는지에 따라 같은 곱셈도 다른 결과를 만듭니다

1 speed → Frequency 증가

Phase가 더 빠르게 바뀝니다. 음량과 공간 간격은 유지됩니다.

2 rings → Period 감소

이미지 안의 반복 간격이 좁아집니다.

3 speaker_level → Multiply 감소

rms로 가는 분기 뒤의 스피커 경로만 작아집니다.

오류 확인과 저장

마지막 5분은 결과 확인과 제출에 사용합니다

값이 안 바뀜 재생 상태 → volume → rms

이미지가 안 바뀜 height의 Brightness 1 참조

화면이 검정 water·surface·camera·lights 경로

소리가 안 들림 audio_out Active와 출력 장치

제출 W03_이름.toe + 결과 캡처 1장
 바꾼 값과 결과의 관계를 설명하는 문장 1개

설정 참고와 추가 설명

기본 실습을 마친 뒤, 필요한 항목을 찾아봅니다

36–45장 노드별 설정 위치·입력값·역할

46–48장 RMS 계산과 Power의 의미

49–51장 Normal Map과 Alpha

52–55장 진폭·중심값·수식·출력 범위

필수 실습은 앞에서 완료했습니다. 설정표는 실습 중에도 참고할 수 있습니다.

tone · volume · rms 설정값

소리 생성, 크기 조절, 분석을 순서대로 확인합니다

노드 / 페이지 / 파라미터	입력값	이 값의 역할
tone / Waveform / Type	Sine	매끄러운 반복 오디오 신호
tone / Base Frequency / Amplitude	110 Hz / 1	음높이 / 원래 진폭
tone / Bias / Phase / Offset	모두 0	추가 편향 없이 시작
tone / Channel / Sample Rate	44100	초당 오디오 샘플 수
tone / Common / Time Slice	On	매 프레임 새 오디오 구간 생성
volume / Mult-Add	Pre-Add 0 / Multiply 0.65 Post-Add 0	입력 신호에 0.65를 곱함
rms / Analyze / Function	RMS Power	입력 구간의 크기 분석

rms Common / Time Slice = Off 유지 · volume을 0.3, 0.8로 비교합니다.

speaker_level · audio_out 설정값

화면에 영향을 주지 않고 스피커로 나가는 크기를 조절합니다

노드 / 페이지 / 파라미터	입력값	이 값의 역할
speaker_level / Mult-Add / Multiply	0.045	청취 경로만 작게 조절
speaker_level / Pre-Add / Post-Add	0 / 0	추가 값 없이 곱셈만 사용
audio_out / Audio Device Out / Active	파일에서는 Off 청취할 때 On	장치로 오디오 보내기
audio_out / Driver / Device	Default / Default	교실 출력 장치에 맞게 선택
audio_out / Volume / Clamp Output	1 / On	출력 배율 유지 / 범위 제한
연결 확인	volume → speaker_level → audio_out	rms는 volume에서 따로 분기

장치를 바꾼 뒤에는 낮은 음량에서 청취 상태를 확인합니다.

mask · Circle TOP 설정값

Circle 페이지에서 영역과 경계, 색을 설정합니다

노드 / 페이지 / 파라미터	입력값	이 값의 역할
Circle / Radius X, Y	0.37 / 0.37	물결을 남길 원의 범위
Circle / Radius Unit	A (Fraction Aspect)	화면 비율을 고려한 반지름
Circle / Center X, Y	0 / 0	현재 기준 중심 위치
Circle / Softness / Unit	0.28 / F (Fraction)	경계에서 높이가 서서히 줄어듦
Circle / Fill Color, Alpha	RGB 1, 1, 1 / Alpha 1	흰 영역은 곱할 값을 유지
Circle / Background Color, Alpha	RGB 0, 0, 0 / Alpha 1	검정 영역은 높이값0으로 제한
Common / Resolution	512 × 512	높이 마스크용 이미지 해상도

이 마스크는 RGB 밝기를 곱합니다. 투명한 배경을 만드는 설정과 구분합니다.

wave_keys · Ramp의 밝기값 표

Table DAT의 열 이름과 세 행을 그대로 입력합니다

노드 / 페이지 / 파라미터	입력값	이 값의 역할
열 이름	pos r g b a	위치 / 색 / 알파
첫 번째 행	0 0 0 0 1	시작은 검정
두 번째 행	0.5 1 1 1 1	가운데는 흰색
세 번째 행	1 0 0 0 1	끝은 다시 검정
rings / Ramp / DAT	wave_keys	이 표를 패턴의 한 구간으로 읽음
rings / Interpolate Notches	Ease-in, Ease-out	표 사이의 밝기 변화를 부드럽게

이 표는 코드가 아닙니다. 위치별로 어떤 밝기를 쓸지 지정한 데이터입니다.

rings · speed 설정값

공간 패턴을 만든 뒤, Phase에 시간 신호를 연결합니다

노드 / 페이지 / 파라미터	입력값	이 값의 역할
rings / Ramp / Type / Period	Circular / 0.22	동심원 형태 / 반복 간격
rings / Extend Left / Right	Repeat / Repeat	범위 밖도 계속 고리 생성
rings / Phase	speed의 phase 채널 참조	고정 숫자 대신 시간 신호 읽기
rings / Output / Combine with Input	Composite and Set Resolution	입력 mask와 합성
rings / Output / Operation	Multiply	mask 밝기와 무늬값을 곱함
speed / LFO / Type / Frequency	Ramp / 0.22	천천히 반복되는 위상 신호
speed / Amplitude / Bias / Play	1 / 0 / On	기본 범위로 재생

speed Channel Name = phase / Sample Rate = me.time.rate (현재 60)

height · height_in · normal_map 설정값

밝기를 만든 후, 같은 이미지를 높이와 법선맵에 사용합니다

노드 / 페이지 / 파라미터	입력값	이 값의 역할
height / Pre / Brightness 1	rms의 chan1 참조	오디오 크기로 밝기를 조절
height / Gamma 1, Brightness 2	1 / 1	추가 보정 없이 유지
height / Common / Pixel Format	16-bit float (RGBA)	높이값의 중간 단계를 보존
height_in / Select / TOP	height	높이 이미지를 재질 쪽에서 읽음
normal_map / Slope / Red, Green	Horizontal / Vertical Luminance	가로·세로의 밝기 차 읽기
normal_map / Blue, Alpha	One / One	각 채널값을 1로 설정
normal_map / Zero Point, Strength	0.5 / 28	차이가0인 기준 / 기울기 강조

normal_map Extend = Hold · 여기서는 Slope가 이미지 경계 밖을 읽는 방식입니다.

surface 내부 · Grid와 Attribute 설정값

표면을 만들고 Normal Map에 필요한 접선 방향을 준비합니다

노드 / 페이지 / 파라미터	입력값	이 값의 역할
grid_mesh / Grid / Connectivity	Quads	사각형 면으로 연결된 격자
grid_mesh / Columns / Rows	201 / 201	변형할 점을 충분히 준비
grid_mesh / Size X, Y / Rotate X	3.4, 3.4 / -90°	정사각형 평면을 눕힘
grid_mesh / Normal, Texture Coordinates	Point / Point	조명과 UV 계산에 사용할 속성
surface_tangents / Attribute Class	Point	각 점에 속성 추가
surface_tangents / Name / Custom Name	Custom / T	접선 속성의 이름
surface_tangents / Type / Components	Float / 4 Value: 1, 0, 0, 1	현재 평면의 접선 방향과 부호

surface Render / Material = water · Render = On

water · Phong MAT 설정값

RGB 페이지에서 색, 반사, 높이 변형을 설정합니다

노드 / 페이지 / 파라미터	입력값	이 값의 역할
RGB / Diffuse	0.67, 0.91, 0.86	표면의 민트색
RGB / Specular / Shininess	0.65, 0.65, 0.65 / 45	반사광의 색과 집중 정도
RGB / Normal Map (Bump) / Bump Scale	normal_map / 1	명암 계산에 표면 방향 사용
Enable Height Map / Height Map	On / height_in	높이로 사용할 이미지 선택
RGB / Channel Source	Red	이미지 R값을 높이로 읽음
RGB / Displace Vertices / Scale	On / 0.35	정점을 실제로 이동시키는 배율
RGB / Displace Midpoint / Parallax Scale	0 / 0	높이0기준 / 시차 표현 끄

개념: 이 예제의 높이 변화량 $\approx (R\text{값} - 0) \times 0.35$

camera · key_light · fill_light 설정값

시점은 고정하고, 따뜻한 주광과 차가운 보조광을 배치합니다

노드 / 페이지 / 파라미터	입력값	이 값의 역할
camera / Xform / Translate	3, 4.8, 5	물결을 비스듬히 내려다봄
camera / Xform / Look At	surface	표면을 향하도록 지정
camera / Projection / Ortho Width	Orthographic / 4.9	원근 변화 없이 일정한 화면 폭
key_light / Xform / Translate	-2, 2.5, 3	주광 위치
key_light / Light / Color / Dimmer	1, 0.94, 0.88 / 0.85	따뜻한 반사광
fill_light / Translate / Dimmer	3, 2, -3 / 0.45	반대쪽 명암 보완
fill_light / Light Color	0.79, 0.87, 1	차가운 보조광

두 Light의 Light Type = Point · 노드 위치와 실제 3D 위치는 별개입니다.

render · out1 · result 설정값

최종 출력 해상도와 참조 경로를 확인합니다

노드 / 페이지 / 파라미터	입력값	이 값의 역할
render / Render / Geometry	surface	그릴 형태 선택
render / Camera(s) / Lights	camera / key_light fill_light	시점과 두 조명 지정
render / Background Color	0.89, 0.90, 0.94, 1	불투명한 연한 배경
render / Anti-Alias	4x	가장자리 계단 현상 완화
render / Common / Resolution	1080 × 1080	최종 이미지 크기
out1 / 입력 와이어	render 연결	부모 컴포넌트 출력으로 전달
result / Select / TOP	out1	마지막 결과를 다시 읽어 표시

out1은 Out TOP, result는 Select TOP입니다. 각각의 역할을 확인합니다.

질문 · 사인파의 평균은 크기일까?

양수와 음수가 번갈아 나오는 소리에서 크기를 읽어봅니다



+0.65 0 -0.65

양수와 음수를 그냥 더하면?

소리는 들리는데 평균이 0이라면?

어떤 값으로 물결 높이를 정할까?

RMS 계산 · 왜 그냥 평균을 내지 않을까?

샘플 4개로 보는 계산 예시 · 그냥 더하면 양수와 음수가 서로 지워집니다

입력값 +0.5 -0.5 +0.5 -0.5

그냥 평균 $(0.5 - 0.5 + 0.5 - 0.5) \div 4 = 0$

① 제곱 0.25 0.25 0.25 0.25

② 평균 $(0.25 + 0.25 + 0.25 + 0.25) \div 4 = 0.25$

③ 제곱근 $\sqrt{0.25} = 0.5$

음수끼리 곱하면 양수입니다. 마지막에는 원래 신호의 크기 척도로 돌아옵니다.

RMS Power · 무엇의 크기를 측정할까?

TD의 메뉴 이름과 스피커의 출력 전력을 구분합니다

TD의 RMS Power: 입력 CHOP 샘플들의 RMS

스피커 전력(W): 디지털 신호만으로 바로 알 수 없음

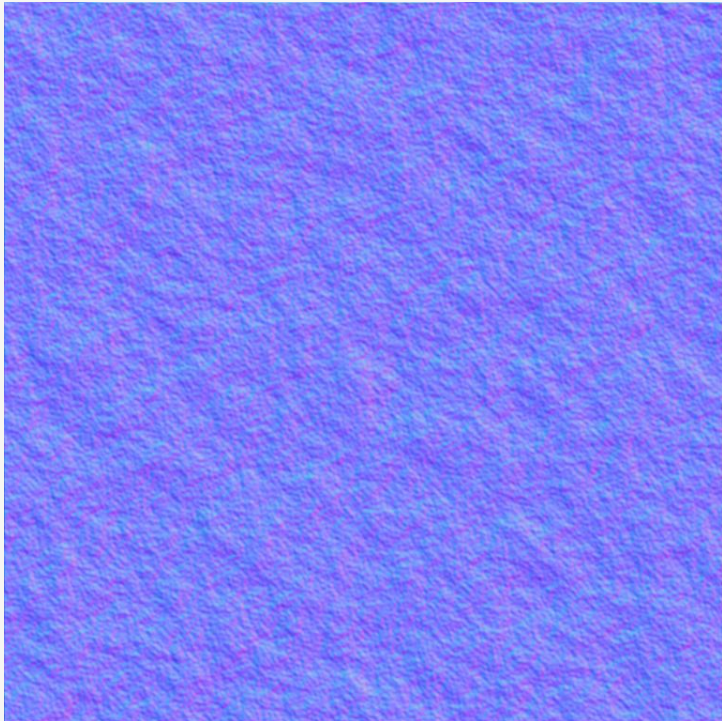
우리가 연결하는 값: 물결 높이를 제어할 신호 크기

입력 진폭이 2배 → RMS 2배 / 평균 제곱값 4배

RMS는 평균 제곱값의 제곱근입니다. 체감 음량과 완전히 같지는 않습니다.

Normal Map은 왜 푸른색일까요?

표면 방향의 세 성분과 RGB의 세 채널을 연결해 생각합니다.



평평한 부분의 방향

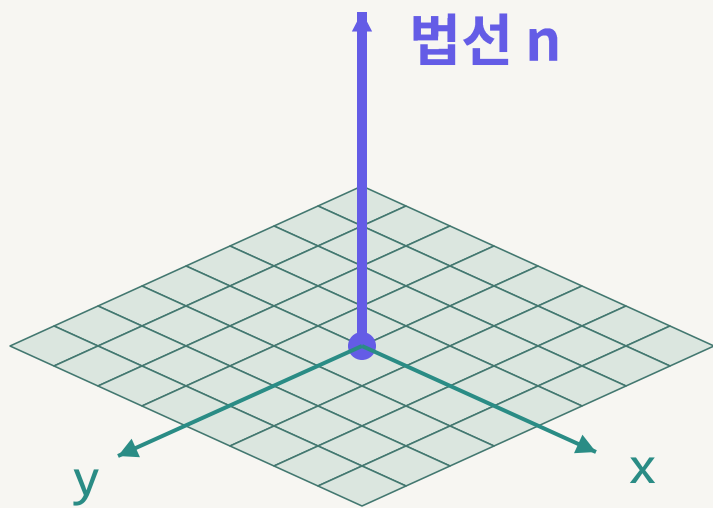
$$\mathbf{n} = (0, 0, 1)$$

X·Y·Z를 R·G·B에 담으려면
음수 값은 어떻게 저장할까요?

먼저 생각하기: 순수 파랑 (0, 0, 1)이 답일까요?

표면의 방향을 RGB에 저장합니다

법선은 표면에 수직인 방향입니다. 이 방향을 색의 세 채널에 나누어 담습니다.



표면 바깥 방향 +z

평평한 면의 법선

$$n = (0, 0, 1)$$

$$\text{RGB} = 0.5 \times n + 0.5$$

R

0.5

G

0.5

B

1.0



연한 파랑

방향 성분 -1~1을 색의 값 0~1로 변환

Normal Map은 빛 계산에 쓸 방향을 바꾸며, 그 자체로 정점을 이동시키지 않습니다.

예측 평평한 면에서 B 채널이 R·G보다 큰 이유는 무엇일까요?

Black과 Zero · 알파값이 다릅니다

둘 다 RGB는 검정입니다. 배경이 보이는지는 Alpha가 결정합니다

Black · 범위 밖 RGBA = (0, 0, 0, 1)



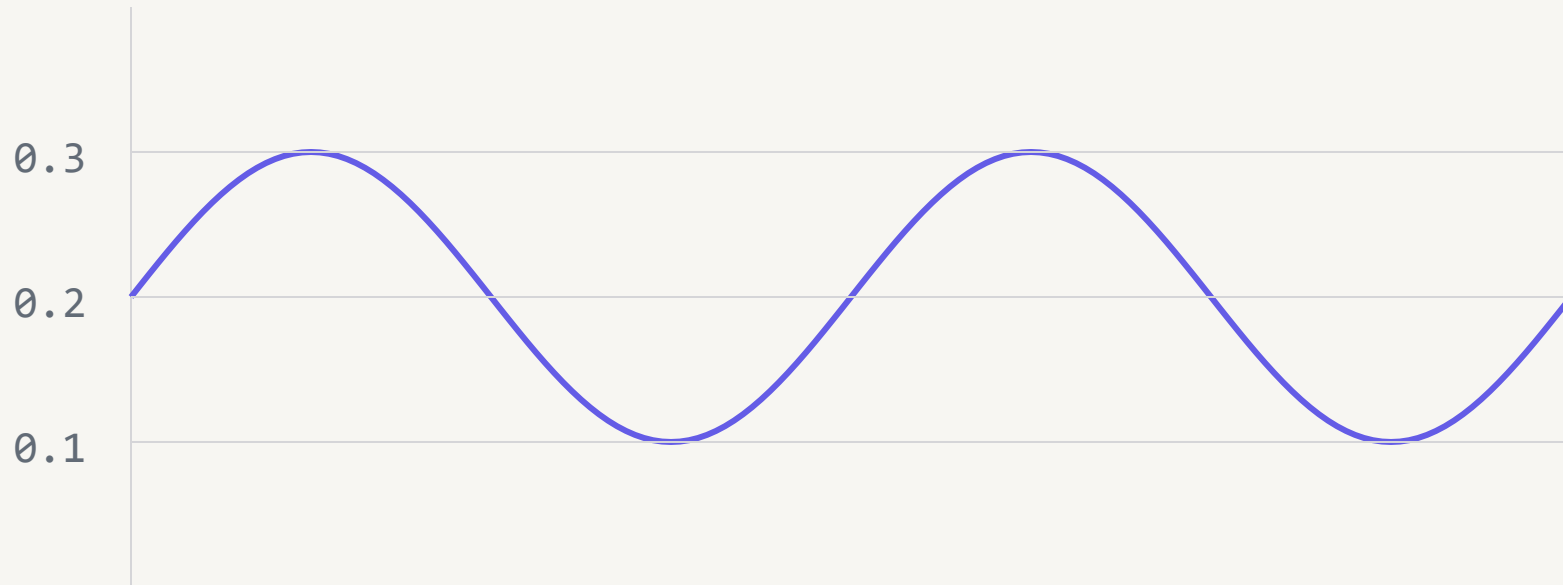
Zero · 범위 밖 RGBA = (0, 0, 0, 0)



Same as Left: 오른쪽도 Extend Left에서 선택한 방식을 사용합니다.

진폭과 중심값

진폭은 중심에서 변화하는 폭이고, Offset은 중심값입니다.



center = 0.2

depth = 0.1

최소 0.1

최대 0.3

중심값 $\pm$ 진폭 = 출력 범위

예측 최소 0.2, 최대 0.4로 옮기려면 center와 depth를 어떻게 설정할까요?

수학식과 실제 TD 입력을 구분하기

LFO가 반복 신호를 만들고, Math가 변화 폭과 중심을 계산합니다.

$$y(t) = \text{offset} + \text{amplitude} \times \sin(2\pi ft + \text{phase})$$

f · 주파수

speed → LFO Frequency

0.5 Hz

amplitude · 진폭

depth → Math Multiply

0.1

offset · 중심

center → Math Post-Add

0.2

phase · 위상

phase → LFO Phase

0 주기

위 식은 개념 표기입니다. TD 입력식에는 $\pi \cdot \times$ 기호를 그대로 붙여 넣지 않습니다.

TD에서는 노드 연산과 파라미터 참조로 같은 관계를 구성합니다.

출력 범위 계산

먼저 계산하고, 다음 장에서 확인합니다.

$$\text{center} = 0.1 \quad \text{depth} = 0.3$$

1 가장 작은 값과 가장 큰 값은 얼마일까요?

2 이 값을 반지름에 쓰면 어떤 문제가 생길까요?

3 반지름 0.1-0.3을 만들 두 값은 무엇일까요?

풀이 · 크기에 맞는 범위 만들기

음수 크기를 피하려면 신호를 연결하기 전에 범위를 정합니다.

문제의 범위

$$0.1 - 0.3 = -0.2$$

$$0.1 + 0.3 = 0.4$$

목표 범위 0.1–0.3

$$\text{center} = 0.2$$

$$\text{depth} = 0.1$$

크기는 $\text{center} - \text{depth}$ 가 0보다 크도록 설계합니다.

Limit CHOP의 Clamp는 범위를 제한하지만 파형의 봉우리가 잘릴 수 있습니다.

노이즈 심화

규칙적인 물결을 이해한 뒤, 불규칙한 값으로 확장합니다

계산 원리와 종류: Value · Perlin · Simplex · fBm

좌표 · Period · Harmonics · Amplitude 비교

같은 값을 Height · Alpha · Color로 사용하기

별도 예제 01_noise_*.tox · 기본 물결 파일과 구분해 열어봅니다.

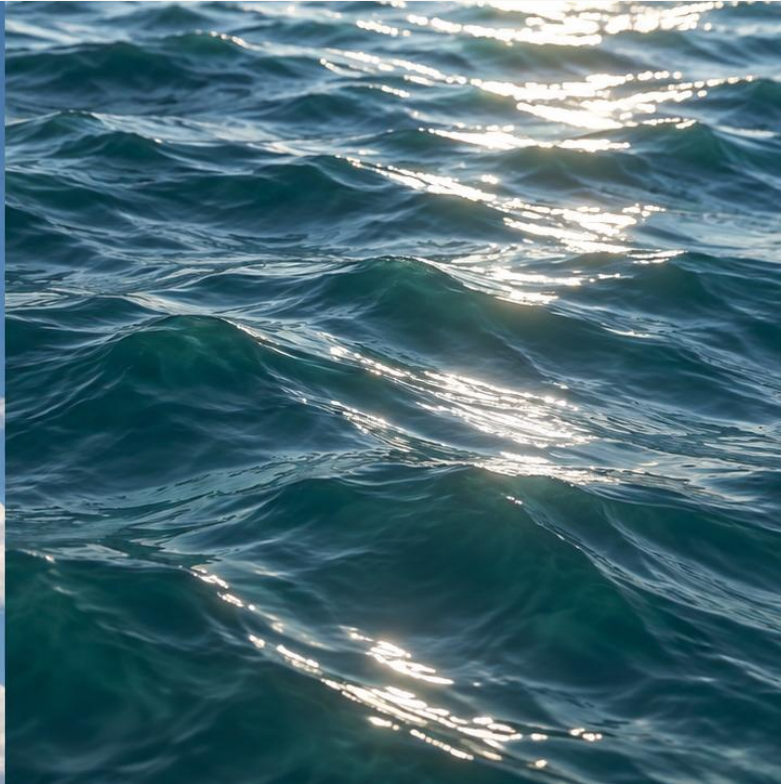
구름·물·지형을 만드는 값의 역할

노이즈에 재질·조명·색을 더하면 자연의 복잡한 모습을 표현할 수 있습니다.

구름의 경계



물의 반사



지형의 높이



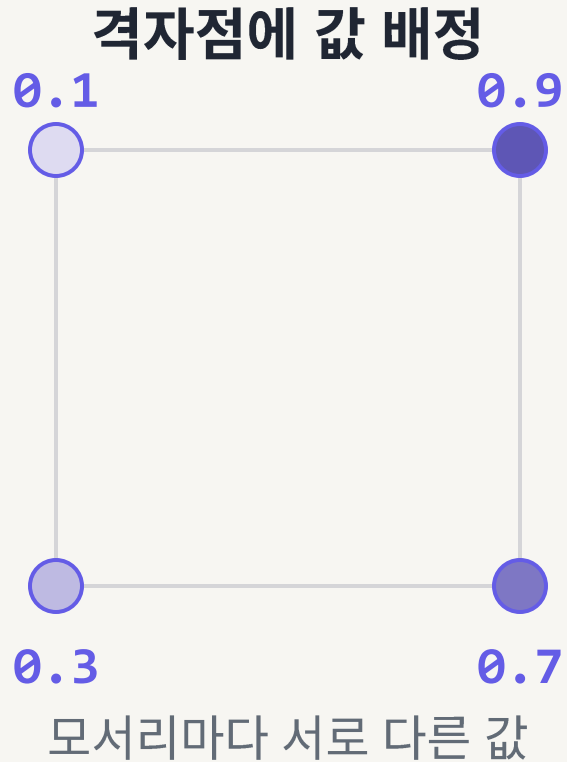
밀도·불투명도

표면 방향·반사

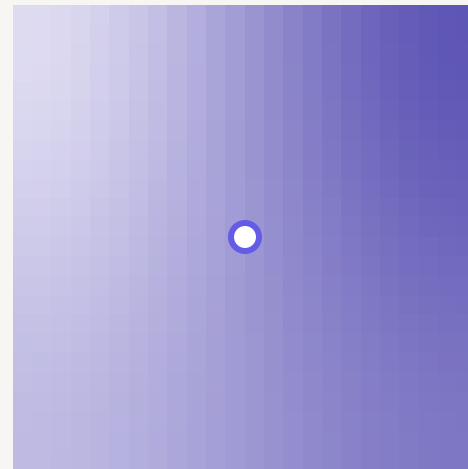
점의 높이·색

노이즈는 좌표마다 계산한 값입니다

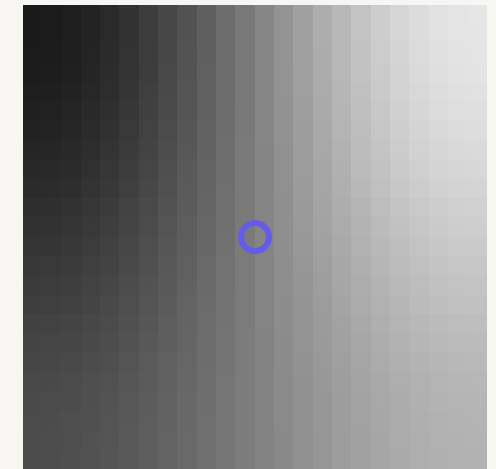
네 모서리의 값으로, 그 사이의 값을 부드럽게 계산합니다.



사이의 값을 보간



값을 밝기로 저장

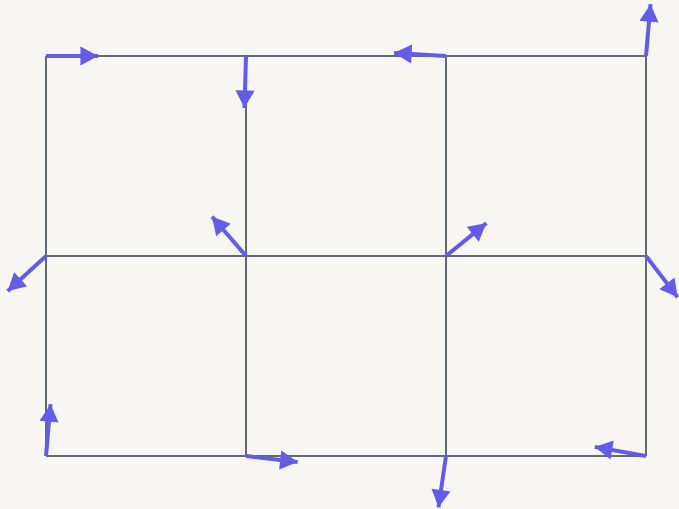


예측 모서리값을 그대로 두고 픽셀 수만 늘리면, 큰 무늬도 달라질까요?

Perlin과 Simplex는 계산 방법입니다

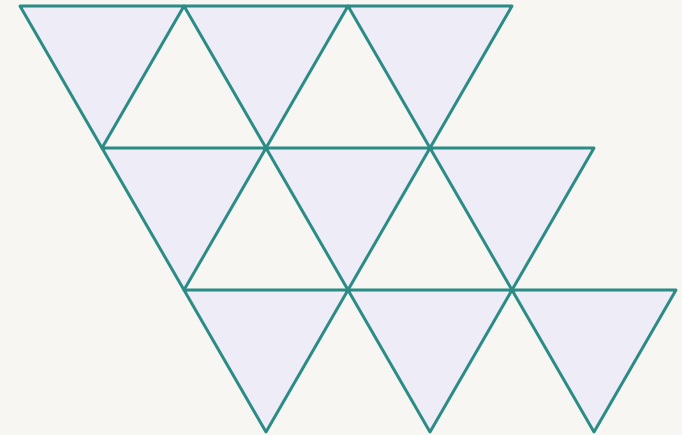
Value는 격자의 값, Gradient Noise는 격자의 방향을 이용합니다.

Perlin · Gradient Noise



격자점의 방향이 주변 값에 기여
기여도를 거리와 함께 부드럽게 합성

Simplex Noise



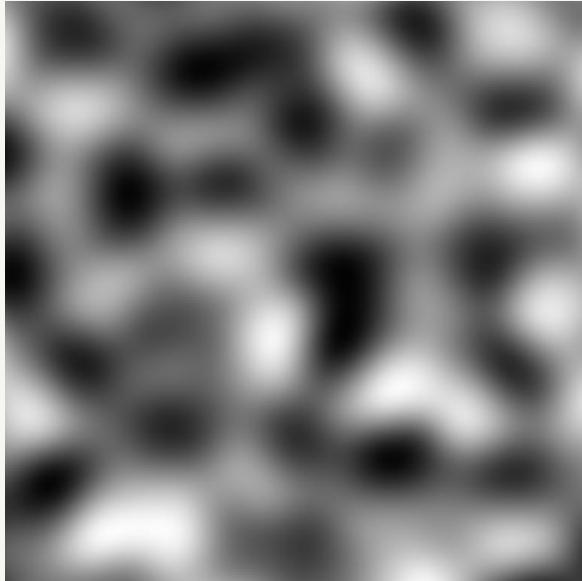
2D 삼각형 · 3D 사면체로 공간 구분
더 높은 차원에서도 계산량을 줄이도록 설계

예측 “Simplex 3D”에서 계산 방법과 입력 차원을 나누어 말해볼까요?

fBm: 크기가 다른 노이즈를 겹치기

큰 덩어리에 더 작고 약한 변화를 더하면 자연스러운 세부가 생깁니다.

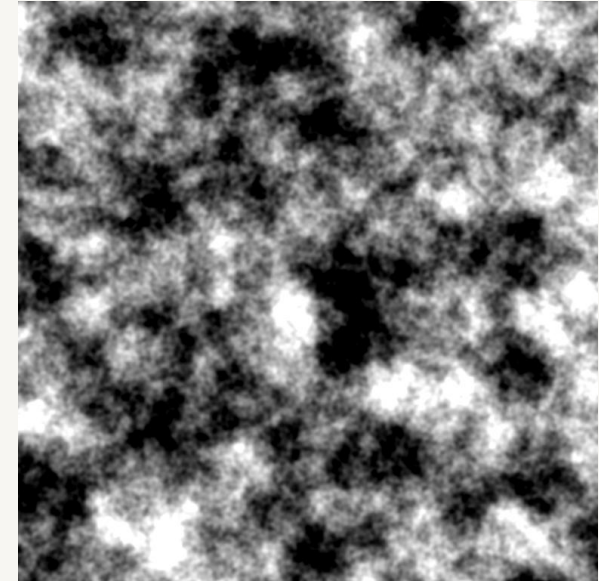
기본 모양



Harmonics = 0

큰 변화
+ 작은 변화
+ 더 작은 변화

작은 세부를 더함



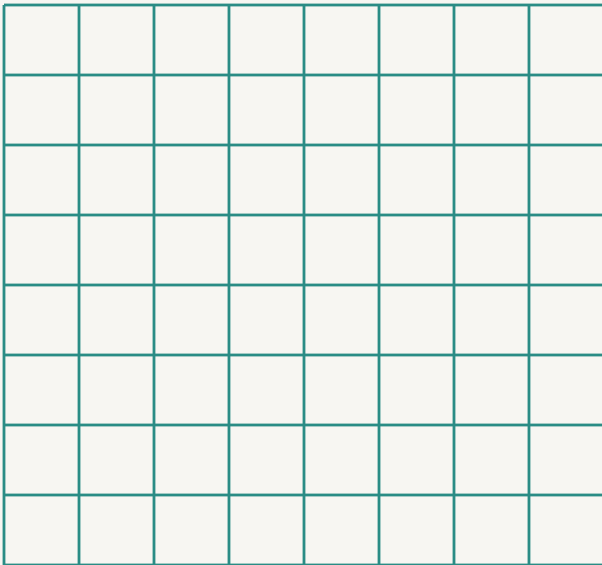
Harmonics = 4

예측 Harmonics를 늘릴 때 큰 덩어리와 작은 요철 중 무엇을 먼저 볼까요?

Domain Warping: 읽는 좌표를 변형하기

높이를 올리는 대신, 노이즈를 읽을 위치를 휘어지게 만듭니다.

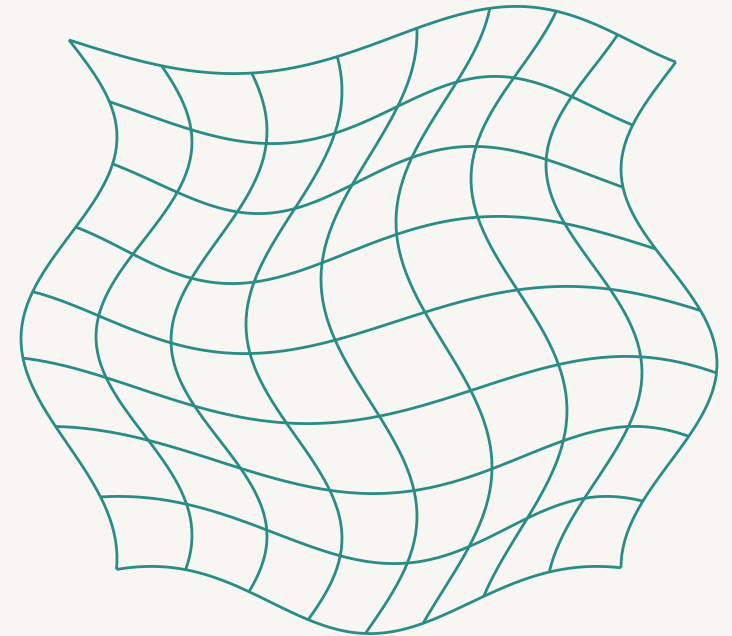
규칙적인 읽기 좌표



좌표에 변위 더하기



노이즈로 바꾼 읽기 좌표



$$N(p) \rightarrow N(p + \text{warp}(p))$$

예측 격자의 z 높이를 바꾼 앞 장과, 이번 장이 바꾸는 대상은 어떻게 다를까요?

2D·3D·4D는 입력 좌표의 수입니다

TOP의 결과가 2D 이미지여도, 계산에는 3개 또는 4개의 좌표를 쓸 수 있습니다.

2D

$N(x, y)$



평면의 두 좌표
한 장의 무늬

3D

$N(x, y, z)$



3차원 값 공간의 한 단면
z를 바꾸면 다른 단면

4D

$N(x, y, z, w)$



네 입력 좌표
w도 별도로 변화 가능

예측 3D Noise를 고르면 자동으로 입체 물체가 생길까요?

Noise TOP의 Perlin · Simplex 계열

계산 방법을 고르고, 움직임에 필요한 입력 차원을 선택합니다.



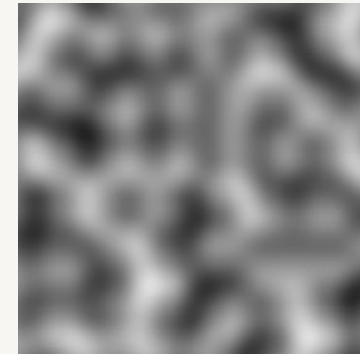
Perlin 2D

입력 x, y
GPU 계산



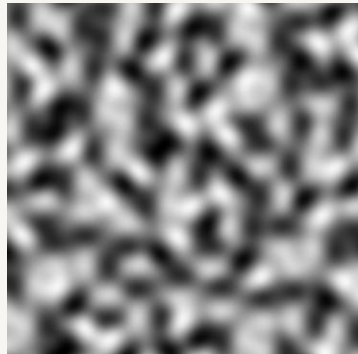
Perlin 3D

입력 x, y, z
GPU 계산



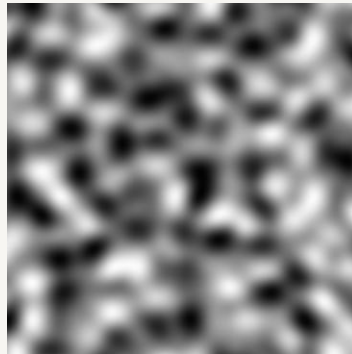
Perlin 4D

입력 x, y, z, w
GPU 계산



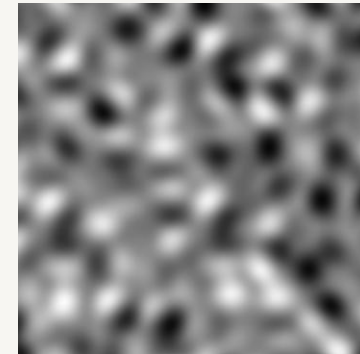
Simplex 2D

입력 x, y
GPU 계산



Simplex 3D

입력 x, y, z
GPU 계산



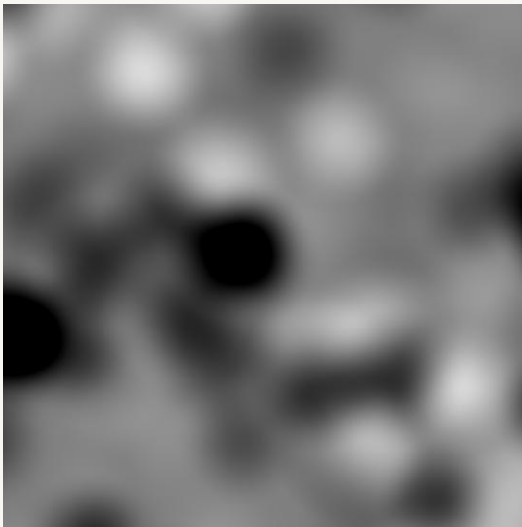
Simplex 4D

입력 x, y, z, w
GPU 계산

Noise TOP의 연속 노이즈 종류

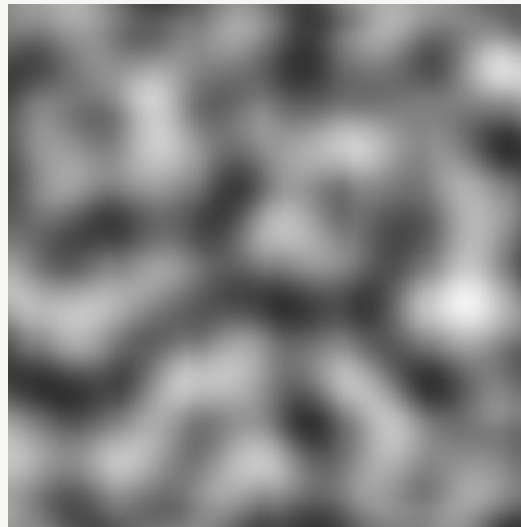
같은 “부드러운 변화”도 계산 방식과 세부 제어가 다릅니다.

Sparse



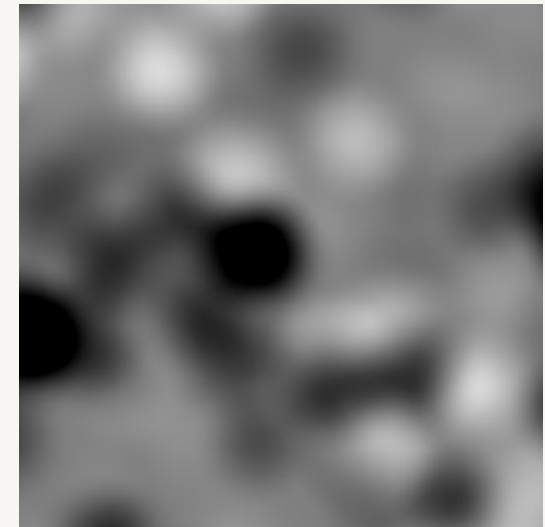
희소한 기여를 합치는 방식
부드럽게 이어지는 변화

Hermite



Sparse보다 계산이 빠른 편
품질과 비용을 비교

Harmomic Summation

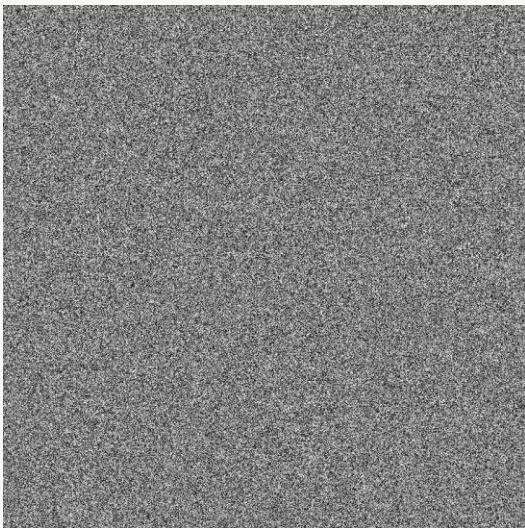


Sparse 계열의 성분 합성
Harmonic Spread로 간격 조절

독립적인 점과 세포 모양

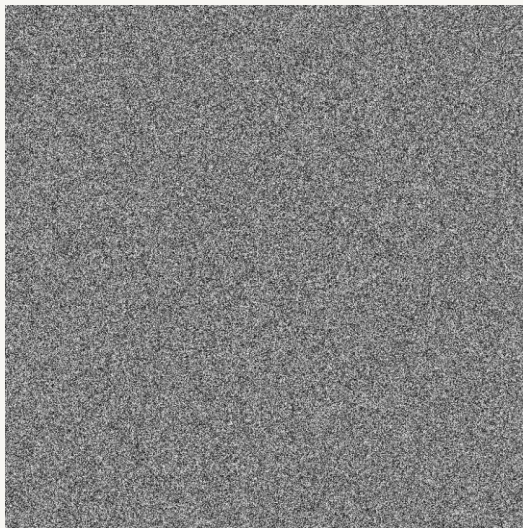
자연스러움을 만들 때도, 원하는 구조에 맞는 출발값을 고릅니다.

Random



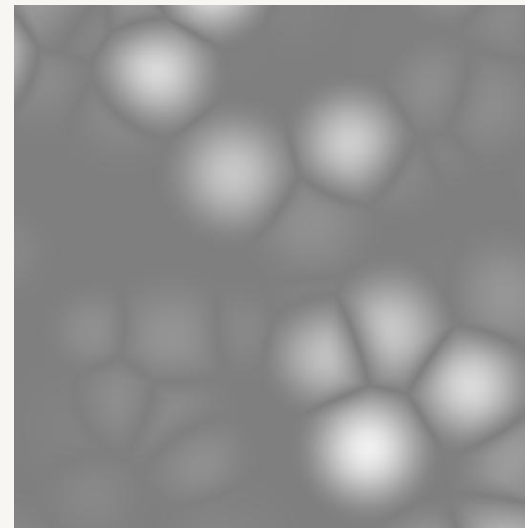
각 표본이 독립적
점·입자에 불규칙성

Random (GPU)



GPU에서 무작위 값 계산
알고리즘·분포는 별도 비교

Alligator



세포 모양의 Cell Noise
갈라짐·덩어리·조직

실습 01 · 노이즈의 전체 연결

시작 파일을 열거나, 두 노드를 직접 만들어 같은 출발점에 도달합니다.



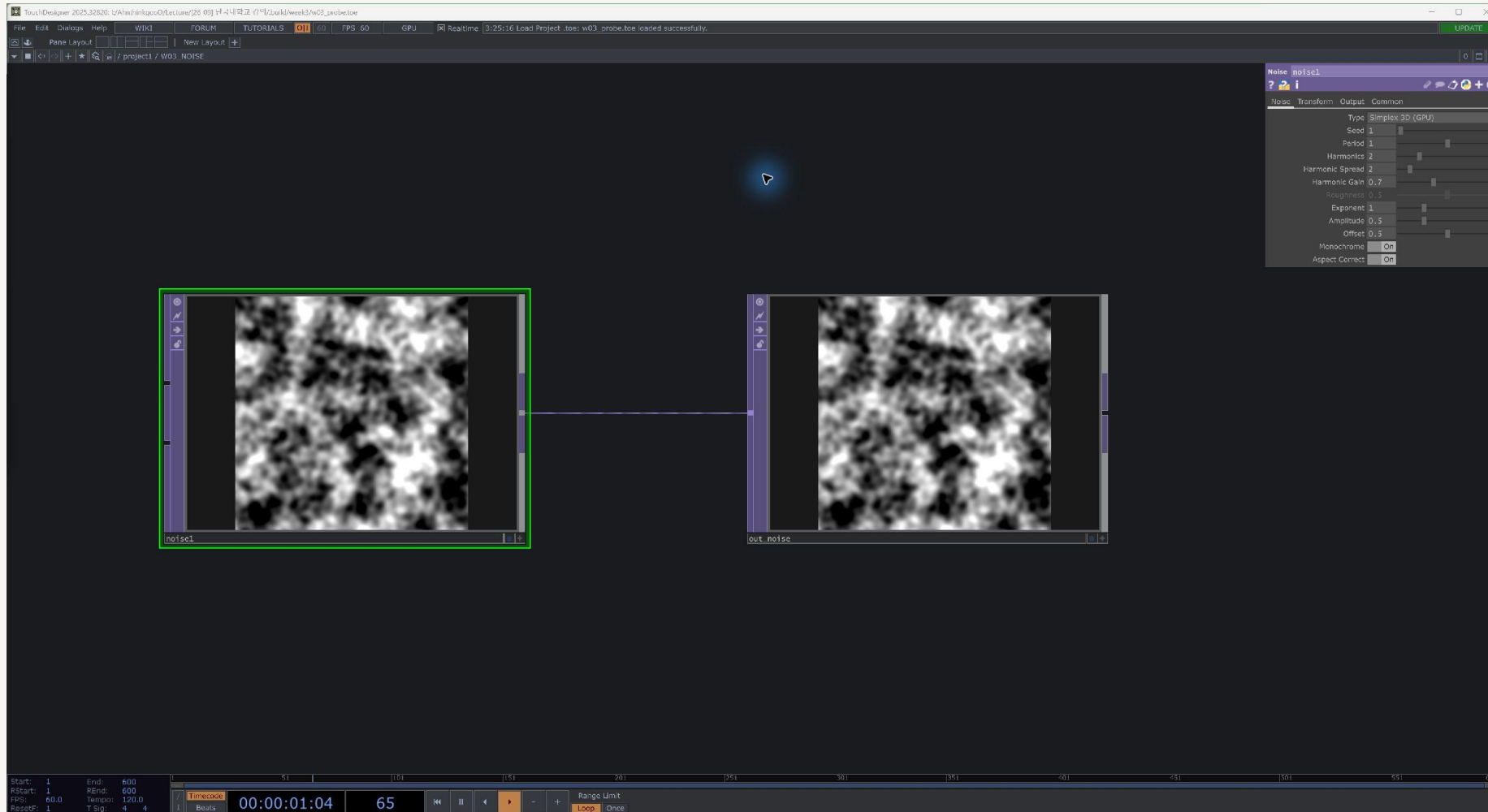
Type: Simplex 3D (GPU)
Seed 1 / Period 1 / Harmonics 0
Amplitude 0.5 / Offset 0.5
Translate X·Y·Z = 0

Tab → Noise TOP 추가
이름을 noise1로 변경
Tab → Null TOP 추가
선 연결 후 out_noise로 변경

noise1 선택 → P → Noise 페이지에서 기준값 설정

Noise TOP 실습의 실제 화면

01_noise_complete.tox



noise1 → out_noise

노이즈 생성 → 출력

캡처: Harmonics 2

비교 기준: 0으로 복원

먼저 두 노드와
연결선을 확인합니다.

Translate X·Y·Z가 바꾸는 것

Simplex 3D에서 한 축만 변경합니다. 픽셀의 위치는 그대로입니다.

기준

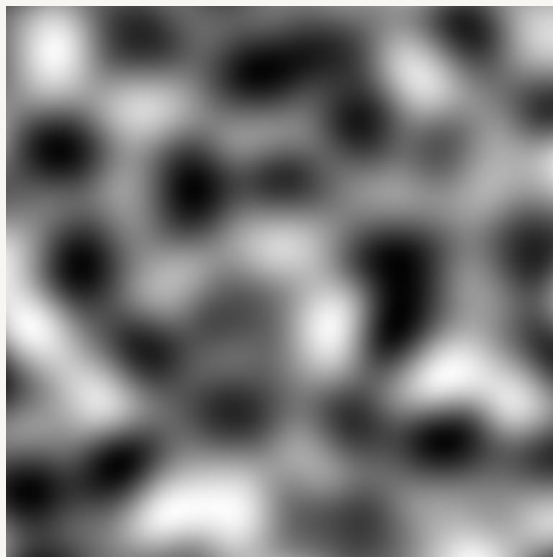
X만 +0.5

Y만 +0.5

Z만 +0.5



X 0 · Y 0 · Z 0



가로 좌표를 바꿔 읽기



세로 좌표를 바꿔 읽기



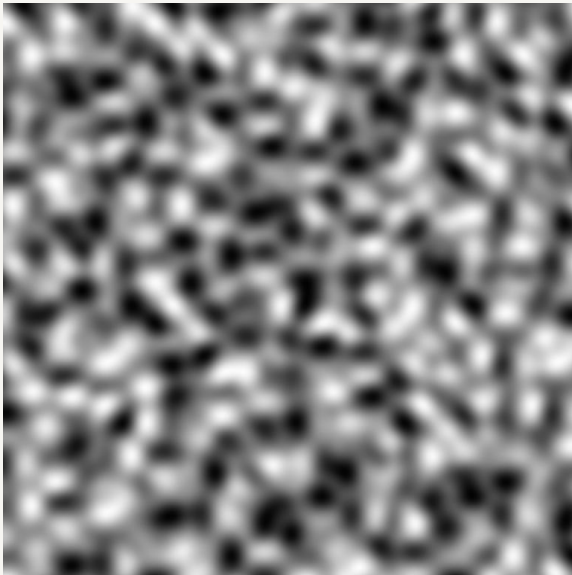
다른 깊이의 단면 읽기

예측 구름이 옆으로 흐르는 변화와, 내부가 일렁이는 변화는 어느 축으로 만들까요?

노이즈의 파라미터 · Period

봉우리 사이의 대략적인 간격을 바꾸는 값입니다.

Period = 0.35



Period = 1



Period = 1.5



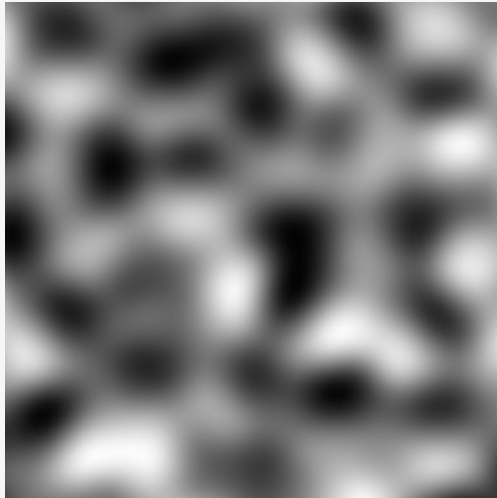
Period가 커지면 무늬의 공간적 크기가 커집니다.

예측 더 작은 얼룩을 많이 만들려면 Period를 어느 쪽으로 바꿀까요?

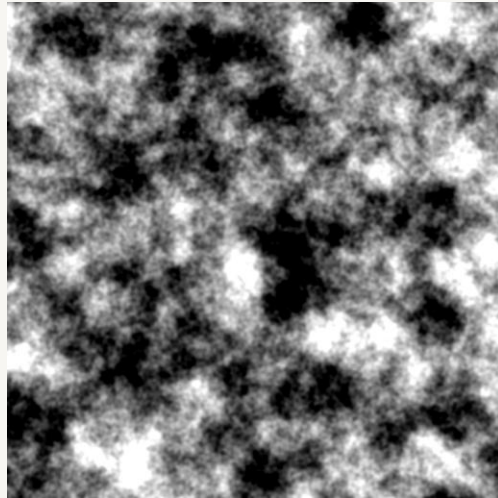
노이즈의 파라미터

Harmonics와 Amplitude 비교

Harmonics 0



Harmonics 4



Harmonics 비교
작은 세부의 추가

Amplitude 0.2



Amplitude 0.5

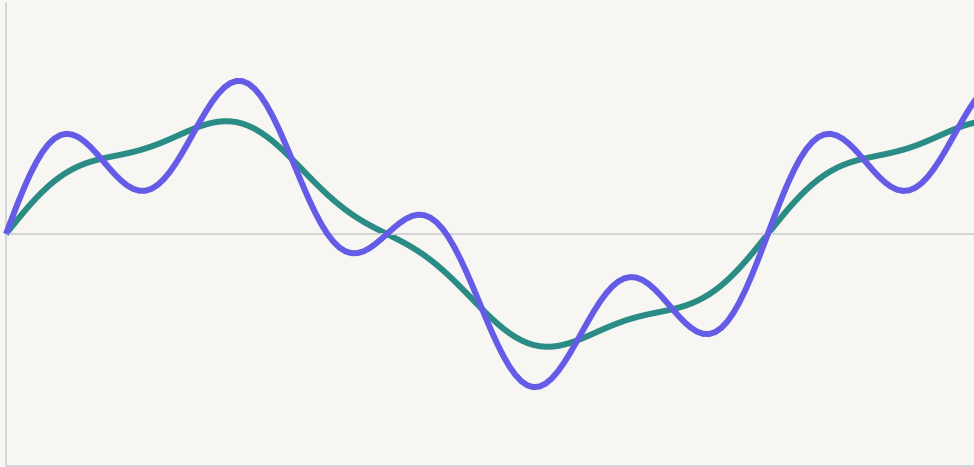


Amplitude 비교
중심에서 값의 변화 폭

노이즈의 파라미터 · 세부 비중과 중심값

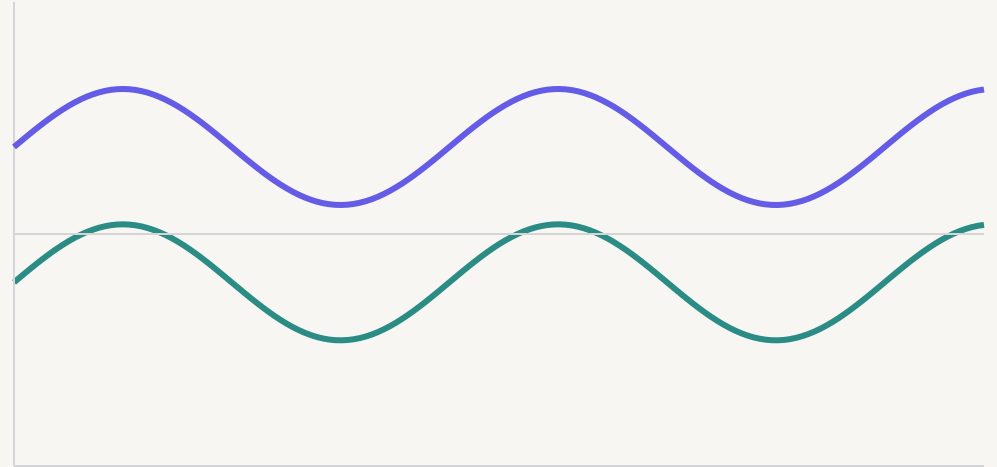
Roughness는 세부의 강도, Offset은 전체 값의 중심을 조절합니다.

Roughness



추가한 고주파 성분의 비중
청록: 작게 / 보라: 크게

Offset



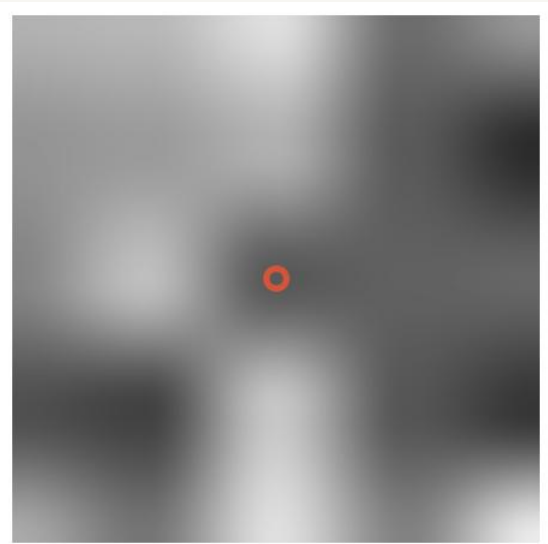
전체 값의 중심을 이동
꼭을 유지하고 위로 옮김

예측 Offset을 올리면 더 밝아질까요, 더 작은 무늬가 생길까요?

샘플링 좌표 이동과 이미지 이동

안쪽 무늬가 비슷하게 움직여도, 경계 밖의 처리 과정은 다릅니다.

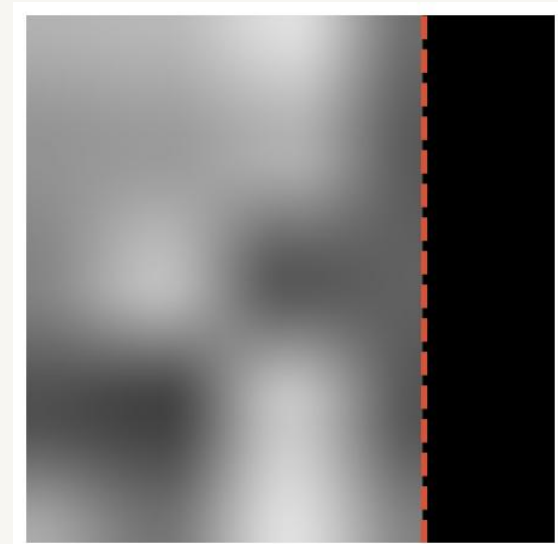
Noise TOP · Translate X



노이즈 함수의 읽는 위치 변경
새 좌표의 값도 계산 가능

새 좌표에서
다시 계산

Transform TOP · Translate X



이미 만들어진 이미지를 이동
경계 밖은 Extend 설정에 따름

Seed는 무늬 선택, 속도는 좌표 변화량

다른 패턴을 고르는 것과 같은 패턴을 빠르게 읽는 것을 구분합니다.

Seed 1



Seed 8



시간과 좌표를 고정하고 Seed만 비교

Seed

다른 무늬를 선택

속도

초당 좌표 변화량

`absTime.seconds * 0.2`

예측 같은 모양을 더 느리게 움직이려면 Seed와 시간식 중 무엇을 바꿀까요?

실습 01 · 한 변수로 의도 만들기

작은 구름, 부드러운 덩어리, 강한 명암 중 하나를 선택합니다.

시작

01_noise_start.tox 또는 두 노드 직접 생성
noise1의 이미지가 보이면 출발

중간

01_noise_checkpoint.tox로 복구 가능
기준값에서 변수 하나만 바꾸고 캡처

완성

01_noise_complete.tox와 비교
변경 전·후 숫자와 의도를 한 문장으로 기록

예측 “더 거칠게”라는 의도를 어떤 파라미터와 비교값으로 설명할까요?

Noise CHOP의 여섯 가지 Type

TOP과 CHOP의 Type 메뉴가 같은 것은 아닙니다.

Sparse · Hermite

부드러운 불규칙 변화
위치·크기 흔들림

Brownian

무작위 변화가 누적되는 형태
떠다니는 궤적에 활용

Harmonic Summation

여러 주파수 성분의 합성
성분 간격 조절

Random · Alligator

독립적인 값 / 세포형 변화
점프하는 값 / 덩어리 감각

CHOP의 Period 단위와 Sample Rate·Time Slice도 함께 확인합니다.

같은 값, 다른 시각 속성

먼저 어느 속성에 연결할지 정하고, 그 속성에 맞는 범위를 만듭니다.

밝기

0.2



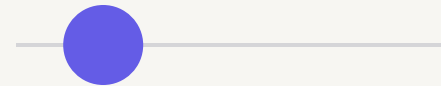
$n \rightarrow \text{RGB}$

크기



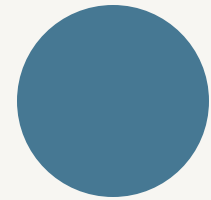
$n \rightarrow \text{반지름}$

위치



$n \rightarrow X \text{ 좌표}$

색



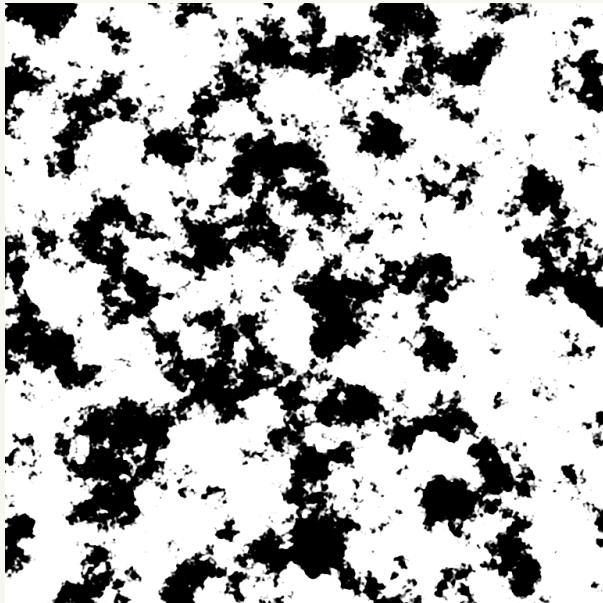
$n \rightarrow \text{두 색의 혼합}$

예측 0.8이 “더 밝다”가 아니라 “오른쪽”이 되는 조건은 무엇일까요?

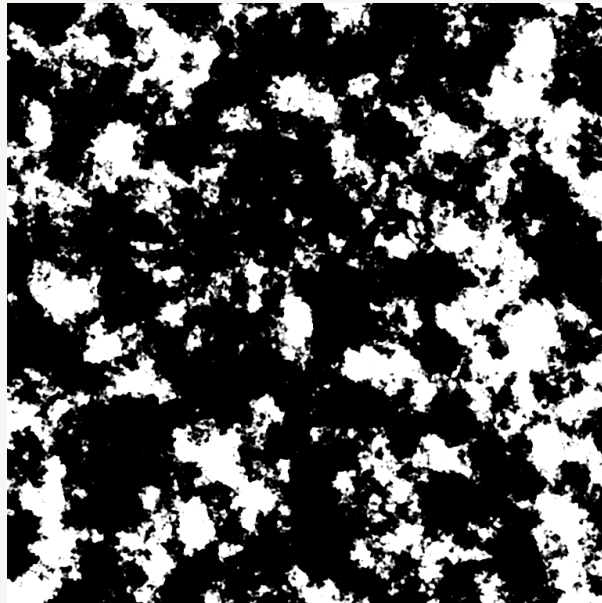
Alpha Map · 남길 영역 선택

흰 마스크를 불투명 영역, 검정 마스크를 투명 영역으로 연결할 수 있습니다.

기준값 0.35



기준값 0.65



$n > \text{기준값}$

남길 영역 = 1

나머지 = 0

Alpha 0: 투명

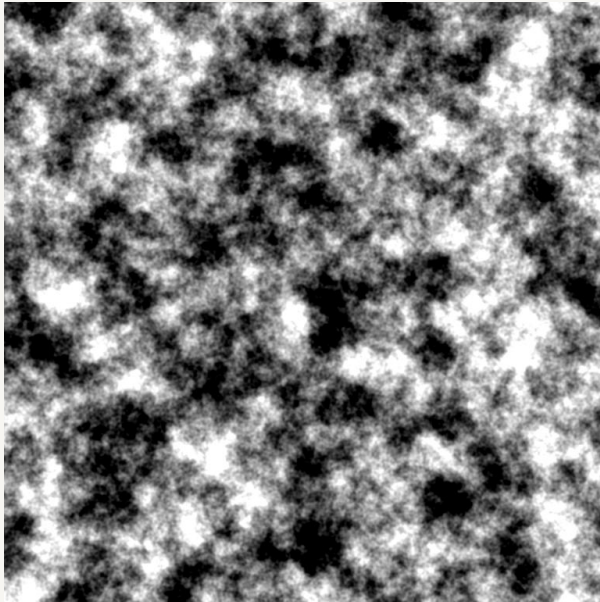
Alpha 1: 불투명

예측 기준값을 높이면 남는 영역은 넓어질까요, 좁아질까요?

Color Map · 값에 색 배정하기

같은 값에 어떤 색을 대응시킬지 정하면 색상 지도가 됩니다.

0-1 값

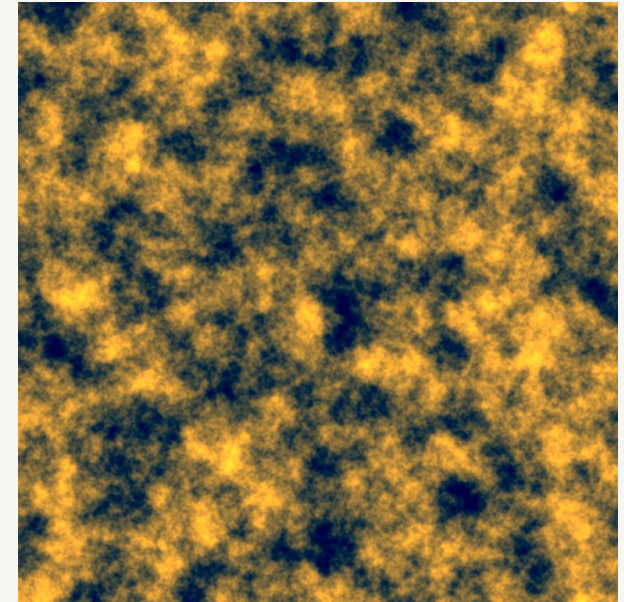


작은 값 → 짙은 청색



큰 값 → 따뜻한 색

색상 범위에 대응



Roughness에 연결하면 반사의 퍼짐, Emission에 연결하면 발광 강도가 됩니다.

사운드 합성과 인터랙션 심화

기본 물결을 완성한 뒤 선택하는 추가 예제입니다

Pattern·Math로 파형 합성하기

소리 크기에 따라 원 움직이기

마우스 입력과 Feedback으로 흔적 남기기

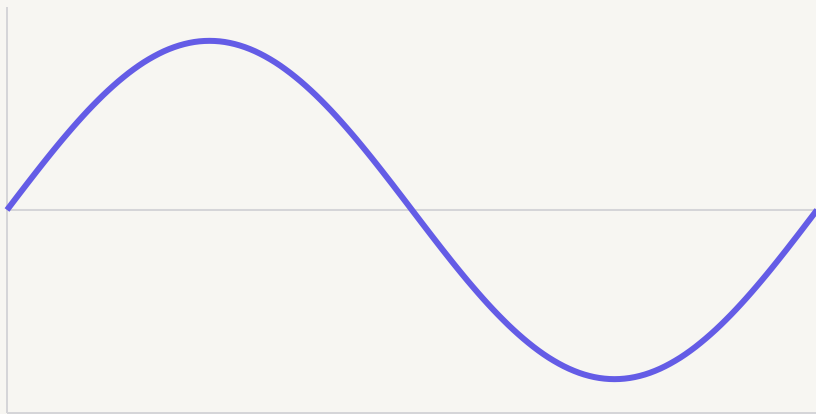
웹캠·공간 선택·그리드 변형으로 확장하기

다른 예제 파일을 사용합니다. 미검증 수동 단계는 각 슬라이드 안내를 따릅니다.

Pattern의 배열을 반복 재생하면 소리가 됩니다

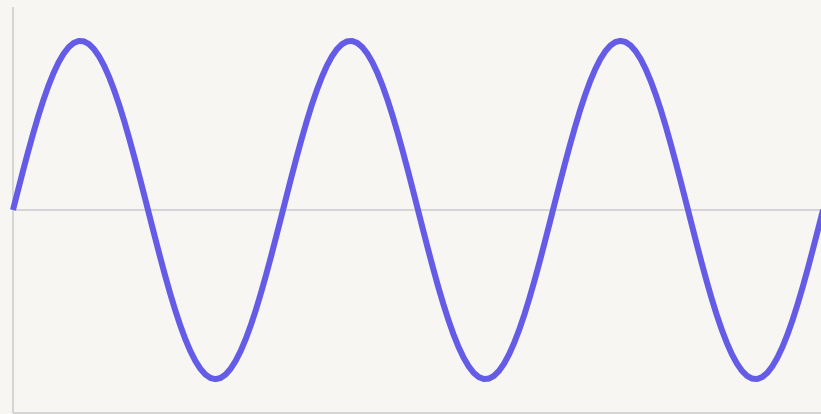
고정된 파형을 만들고, Audio Oscillator의 Playback Source로 읽습니다.

Pattern · 한 주기의 배열



240 samples / 48,000 samples/s

player · 반복해서 읽기

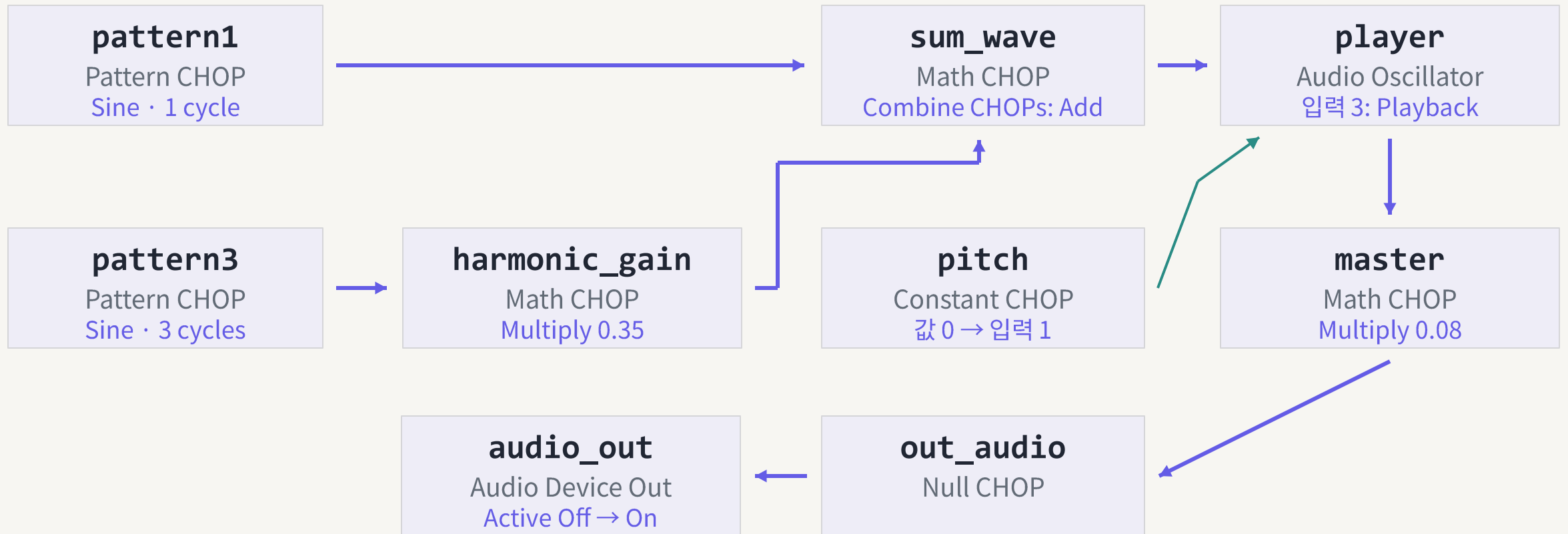


$48,000 \div 240 = 200 \text{ Hz}$

예측 같은 샘플률에서 한 주기의 길이를 480으로 늘리면 음높이는 어떻게 될까요?

Pattern · Math 사운드의 전체 흐름

한 주기에 배음을 더하고, 합성한 파형을 반복해서 재생합니다.



사운드 실습 · 단일 파형부터 확인하기

노드 이름과 입력 순서를 확인하며 단계별로 완성합니다.

시작 · Pattern

Type Sine / Length 240 / Number of Cycles 1

Amplitude 1 / Offset 0 / Sample Rate 48000

중간 · 재생

pattern1 → player의 세 번째 입력 (Playback Source)

pitch Constant=0 → 첫 입력 / 둘째 입력 비율

완성 · 합성

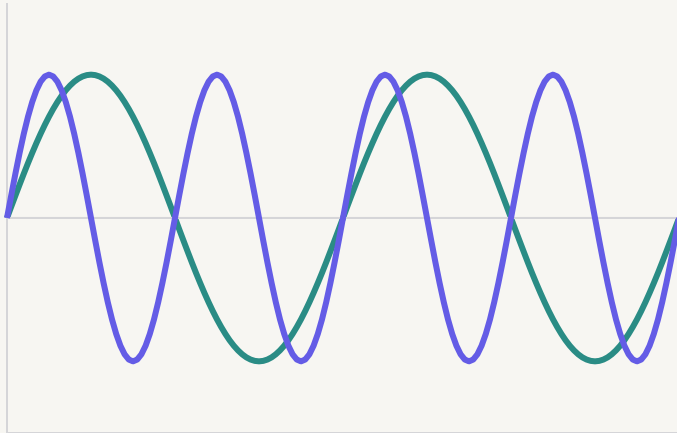
pattern3 Cycles=3 → Math $\times 0.35$ → sum_wave에서 Add

player → master $\times 0.08$ → out_audio → audio_out

음높이 · 음색 · 음량을 구분하기

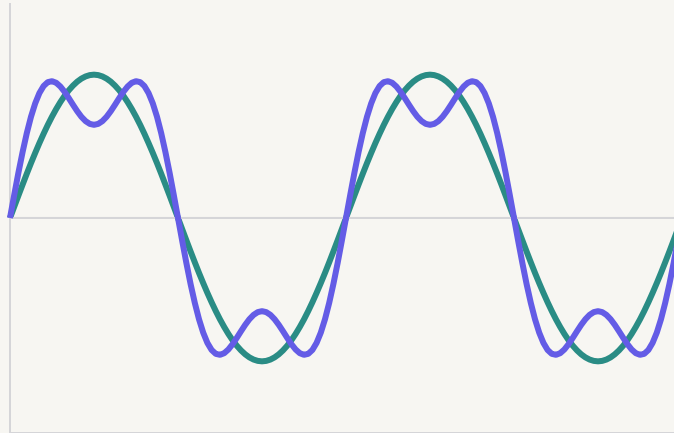
한 번에는 Pitch, 배음 비중, Master Gain 중 하나만 바꿉니다.

Pitch



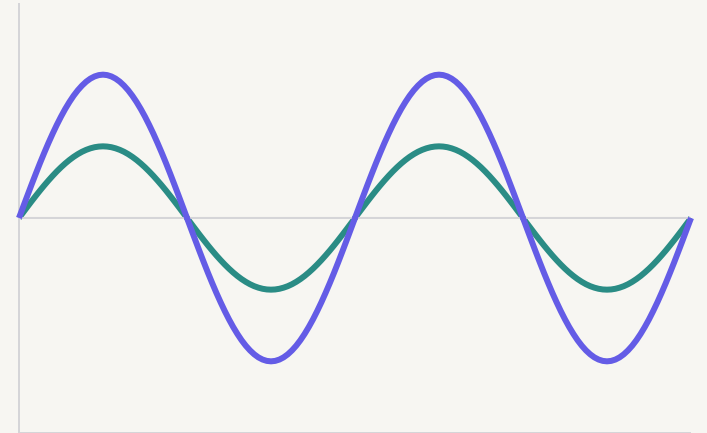
0 → 1: 재생 속도 2배

배음 비중



0 → 0.35: 파형 모양 변화

Master Gain

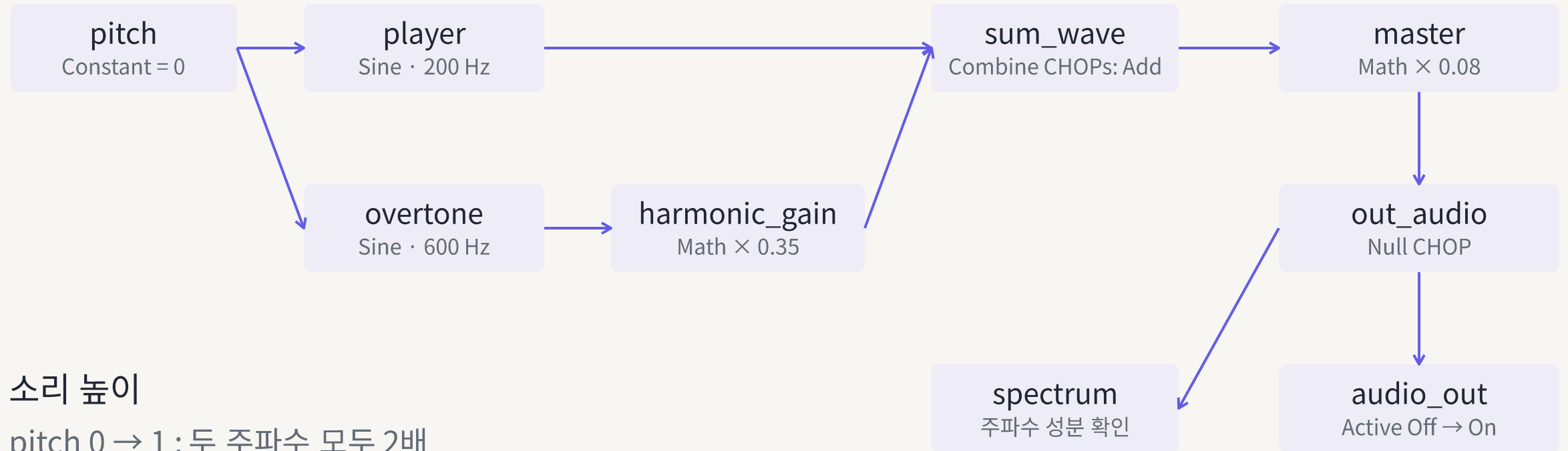


0.04 → 0.08: 진폭 2배

예측 200 Hz를 유지하면서 소리의 질감을 바꾸려면 무엇을 조절할까요?

두 사인파를 더해 소리 만들기

08_sound_native_complete.tox · Audio Oscillator + Math 실행 예제



소리 높이

pitch 0 → 1 : 두 주파수 모두 2배

음색

harmonic_gain 0 → 0.35 : 600 Hz 성분 추가

빠른 파형에서 느린 음량값 읽기

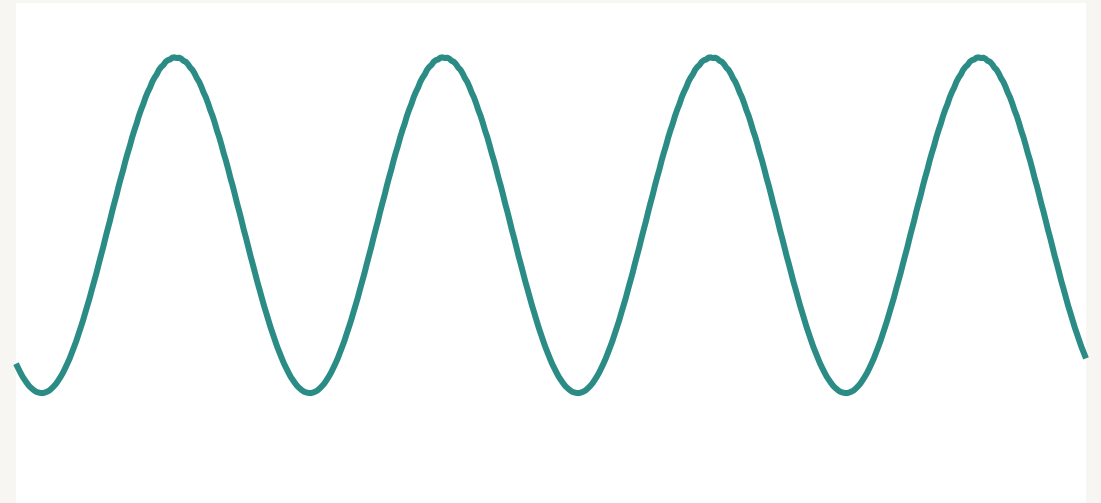
같은 TD 녹화 파일에서 측정 · 순간값과 RMS는 다릅니다

오디오 파형 · 0.03초 확대



0을 중심으로 빠르게 + / - 반복

RMS · 8초 동안의 변화



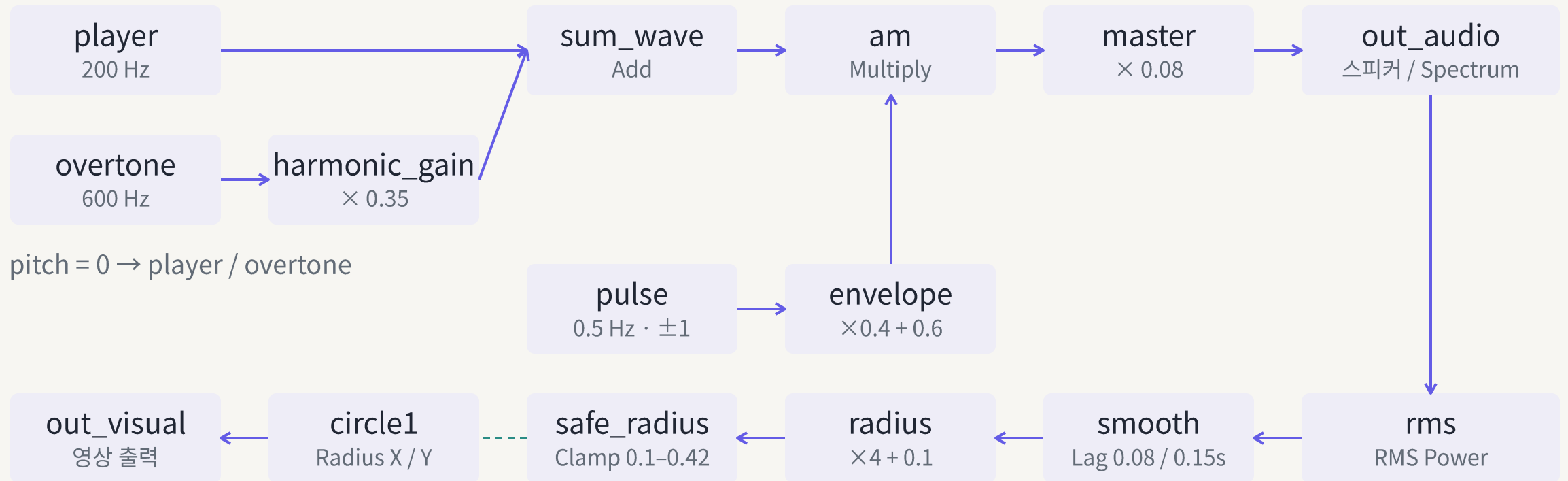
Analyze → Function: RMS Power

예측: 일정한 사인음을 틀면 원도 계속 커졌다 작아질까요?

음량이 일정하면 RMS도 거의 일정합니다. 느린 진폭 변화가 필요합니다.

소리의 음량으로 원의 크기 제어하기

09_audio_reactive_complete.tmx · 실선 연결 / 점선은 파라미터 참조

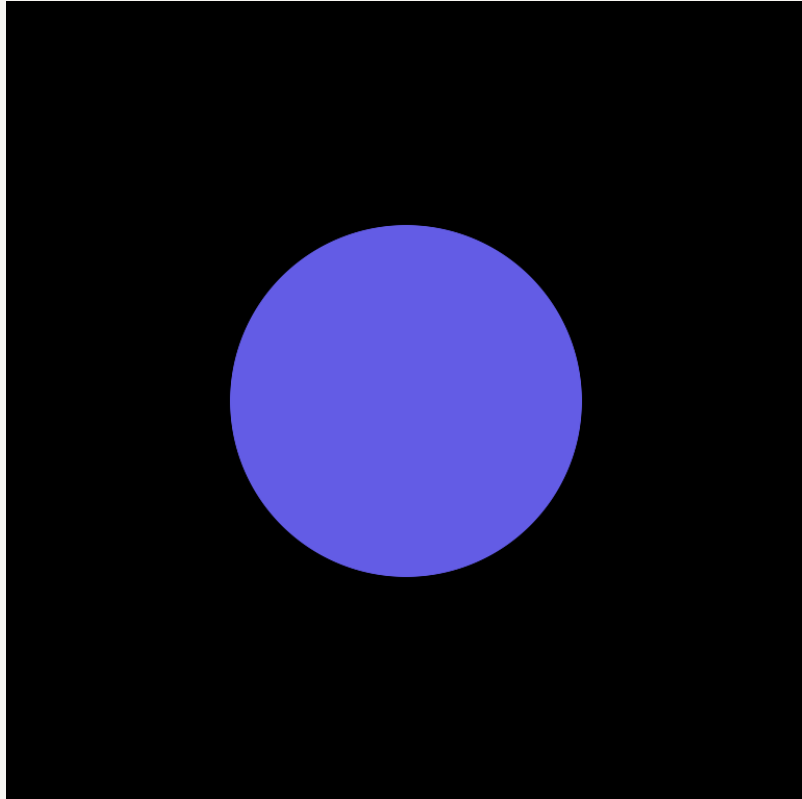


envelope는 0.2-1.0 → 소리의 진폭이 천천히 변합니다.

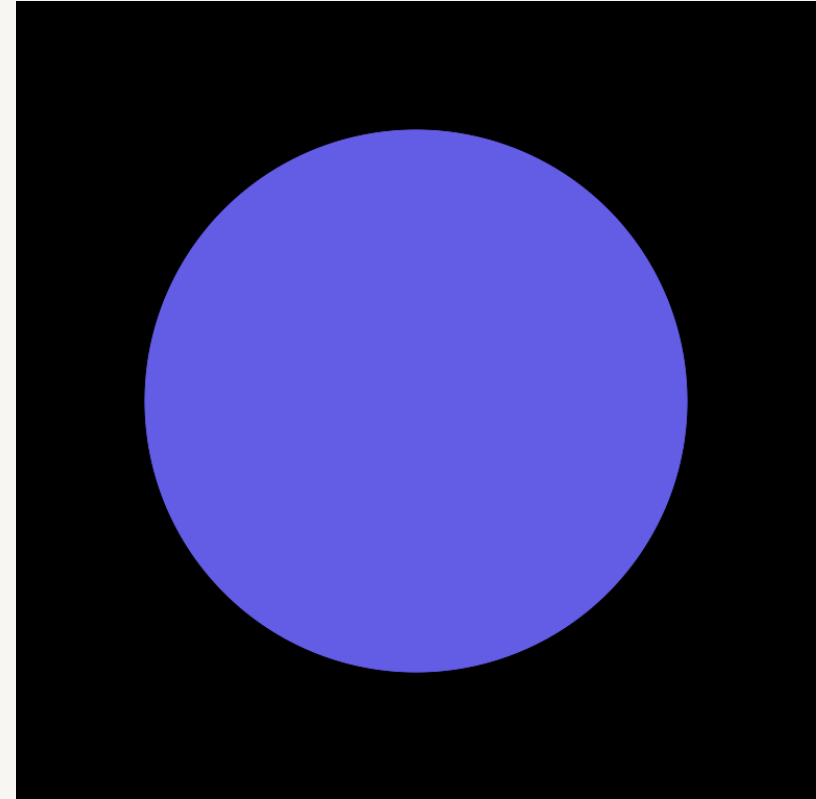
원은 LFO를 직접 읽지 않고 out_audio의 RMS를 읽습니다.

같은 소리에서 음량만 바꾸면?

실제 TOP 출력 비교 · envelope = 1 고정 · Lag 우회 / 나머지 조건 동일



master $\times$ 0.04
RMS 약 0.030



master $\times$ 0.08
RMS 약 0.060

누른 곳에서 소리와 원이 반응합니다

10_feedback_start.tox → 10_feedback_complete.tox

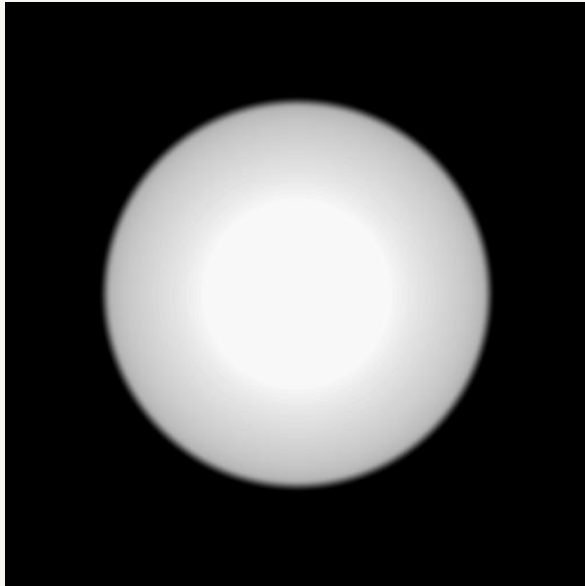


input_panel의 Viewer Active를 켜고 누른 채 드래그

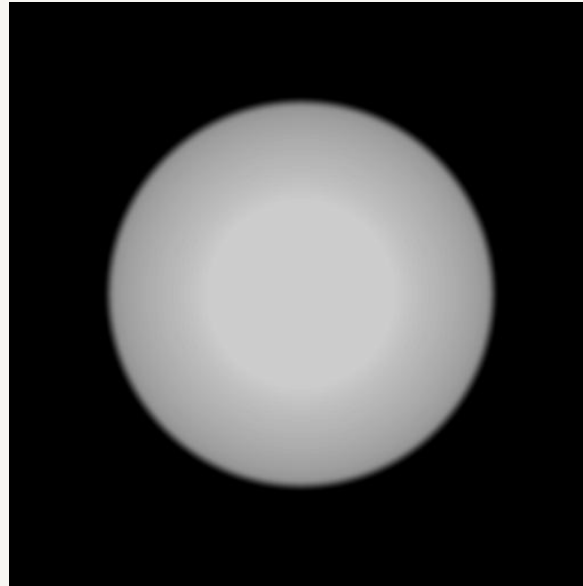
예측: 버튼을 놓으면 소리와 흔적이 동시에 사라질까요?

소리의 여운과 화면의 잔상을 나눕니다

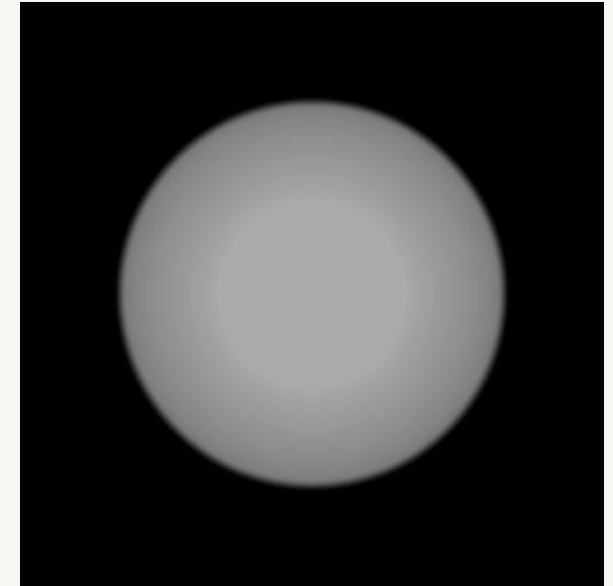
입력을 놓아도 기억된 마스크는 자체 속도로 줄어듭니다.



입력 해제 직후



29프레임 뒤



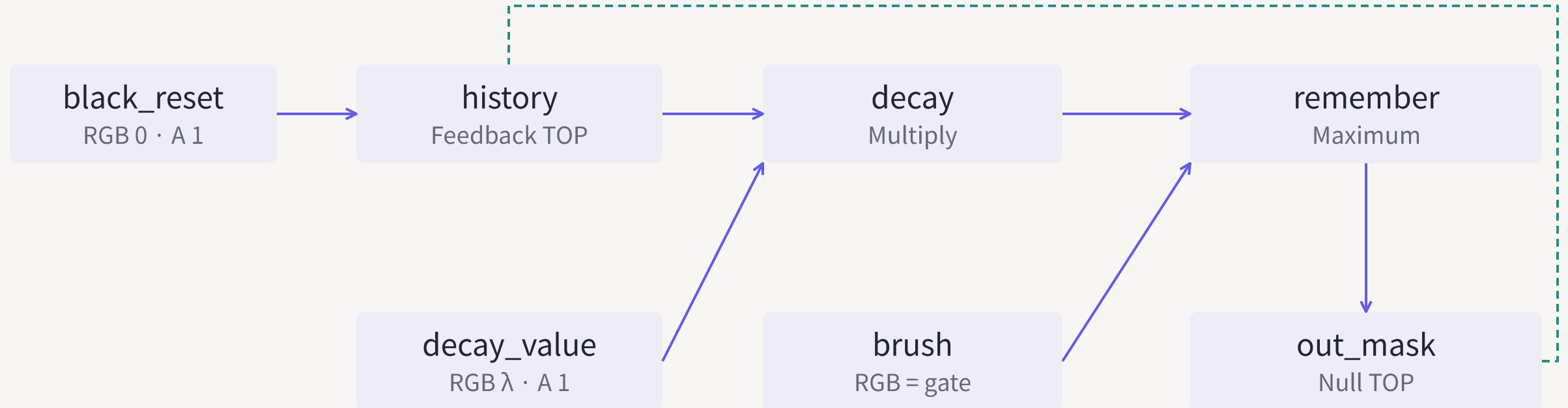
57프레임 뒤

실제 TD 출력 · 테스트 입력 1 → 0 · 같은 위치

Lag: 소리의 release / Feedback: 이미지의 decay

Feedback은 직전 결과를 다시 읽습니다

현재 원과 어두워진 과거를 Maximum으로 합칩니다.



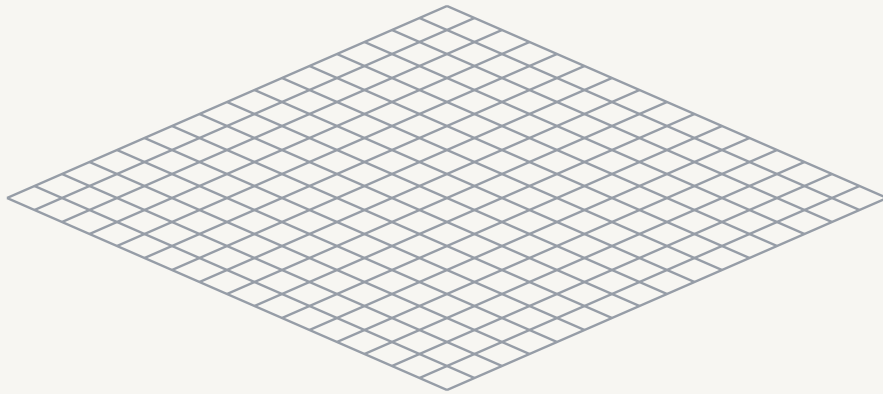
history → Target TOP = out_mask

$\lambda = 0.98566$: 60 FPS 기준 약 0.8초마다 밝기 절반

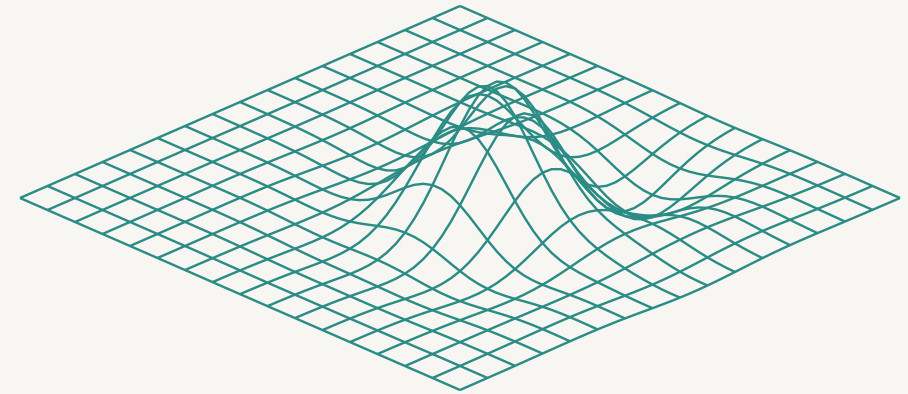
decay_value 색 공간: Passthrough · 16-bit float · Reset으로 지우기

마스크가 있는 곳만 높이를 바꿉니다

$$\text{height} = \text{mask} \times (2 \times \text{noise} - 1) \times 0.4$$



mask = 0 → 원래 평면



밝은 영역 → 더 큰 변형

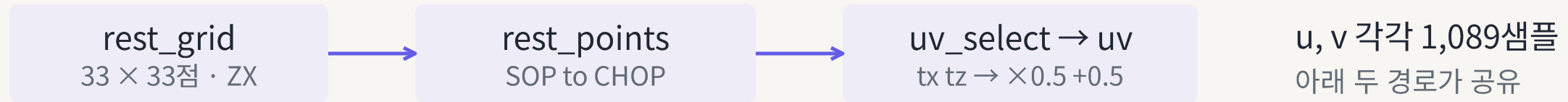
개념 도식 · 실제 SOP 렌더 캡처가 아닙니다.

문제: $(\text{mask} \times \text{noise} - 0.5) \times 2$ 는 왜 틀릴까요?

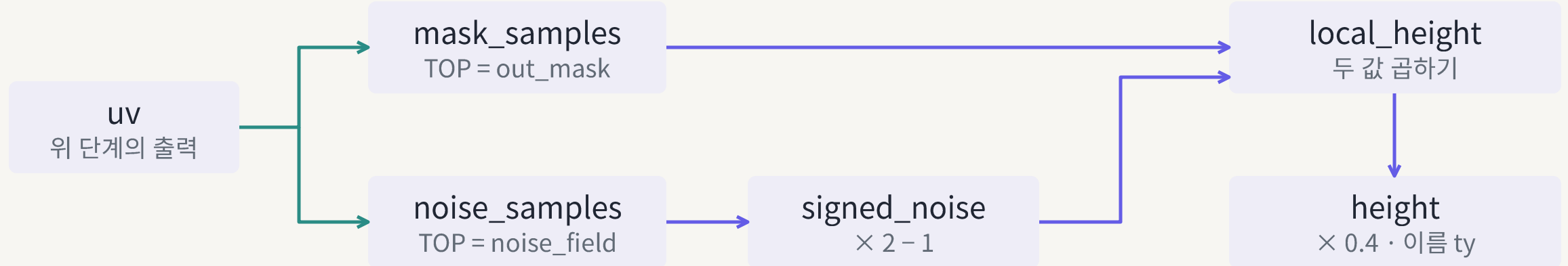
같은 좌표에서 마스크와 노이즈 읽기

두 이미지의 값을 같은 점에 대응시켜 높이를 계산합니다.

① 점의 좌표 준비



② 같은 UV로 샘플링 → 값 계산



final_points: 원래 tx · 새 ty · 원래 tz를 Merge

CHOP to SOP: final_points → P · One to One · Compute Normals

웹캠은 움직인 정도를 energy로 만듭니다

선택 확장 · 카메라 입력과 실제 동작은 아직 실행 검증 전



정지 2초 → 잡음 관찰 → 손을 움직여 범위 조절

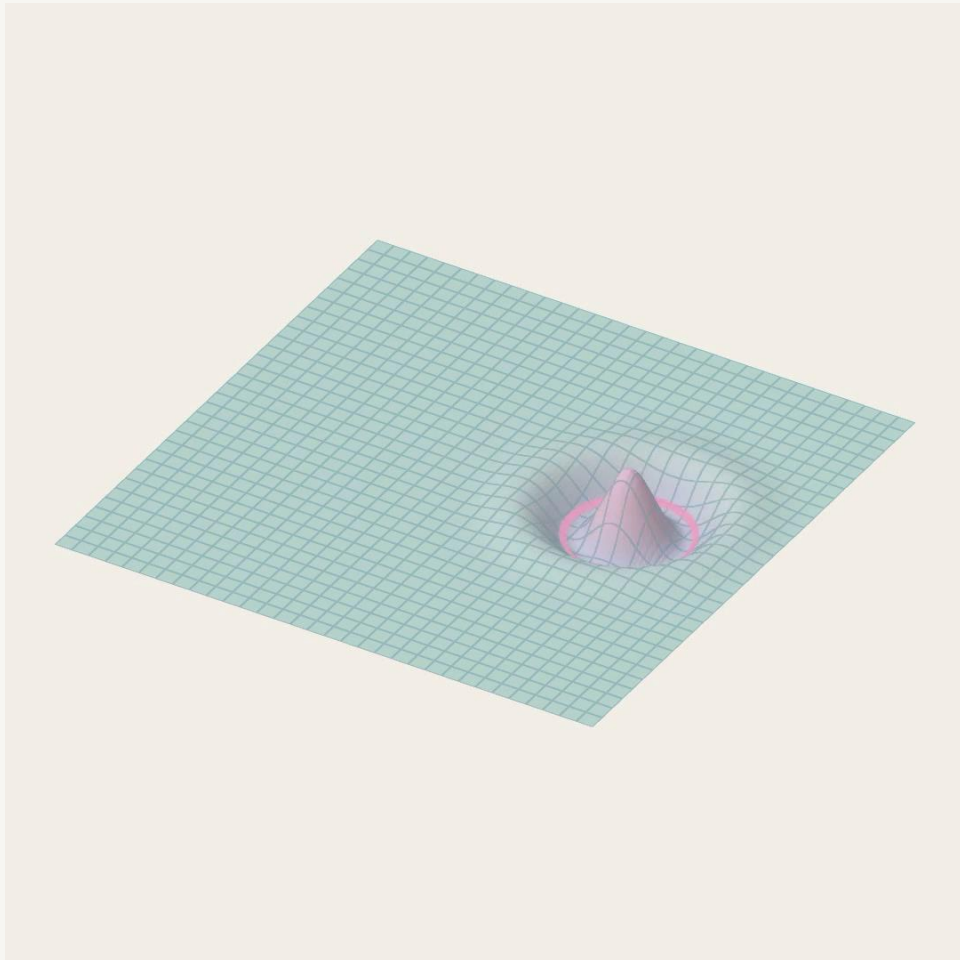
평균 밝기만 읽으면 움직임 대신 방의 밝기에 반응합니다.

기본: 중앙 원의 에너지 / 심화: 움직인 영역의 중심 위치

선택 과제: 마우스와 웹캠이 같은 energy 범위를 내도록 맞추기

선택한 곳에서 소리와 그리드가 진동합니다

11_pastel_spatial_oscillator_complete.tox · 실제 출력 1080 × 1080



위치

panel 안의 그리드를 누른 채 이동

소리

200 Hz + 600 Hz · master 0.08

움직임

2 Hz 반복 · 소리 RMS로 변화 폭 제어

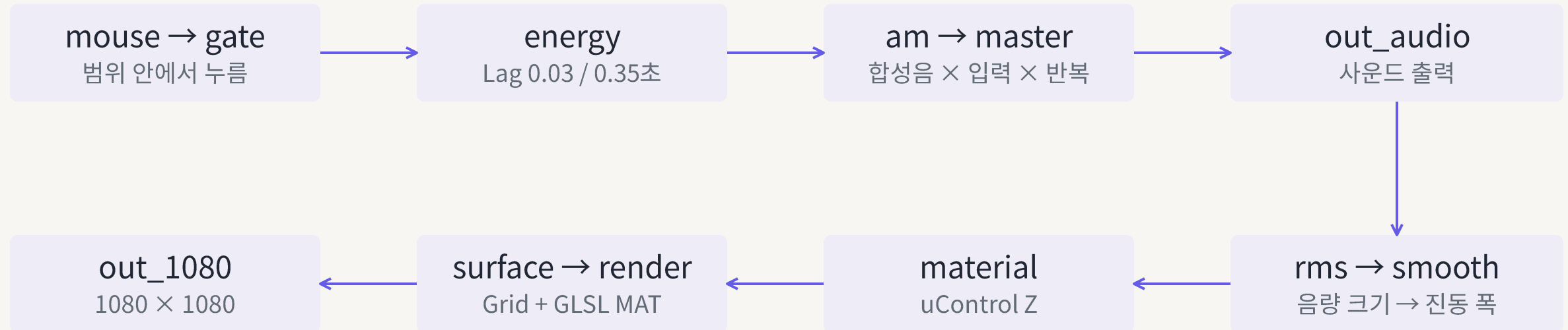
해제

입력 0 → 소리와 변형이 줄어듦

영상: W03_pastel_spatial_1080.mp4

위치·음량·반복값을 따로 연결합니다

제공된 그리드와 셰이더에 CHOP 제어값을 연결하는 실습



mouse u / v → uControl X / Y : 선택 위치

pulse chan1 → uControl W : 2 Hz 반복값

시작: 높이 0 → 중간: RMS 연결 → 완성: 반복값 연결

수업 후 참고 영상

영상 제목 아래 링크를 클릭하면 해당 자료가 열립니다.

Inigo Quilez · I painted a Landscape with Mathematics

00:48 기본 노이즈 / 04:12 fBm / 13:53 하늘 / 34:48 구름

[YouTube 영상 열기](#)

Lake Heckaman · Mastering GLSL, Lesson 8: FBM Noise

노이즈를 여러 규모로 합성하는 과정 · GLSL 구현은 선택 학습

[YouTube 영상 열기](#)

TouchDesigner · Electronic Music Studio — Owen Kirby

03:30 오디오 합성 시작 · 긴 워크숍 중 필요한 부분만 시청

[YouTube 영상 열기](#)

개념과 파라미터를 다시 확인할 곳

공식 문서와 원리 설명을 함께 읽으면, 값을 바꾸는 이유를 이해할 수 있습니다.

The Book of Shaders · 11 Noise / 13 fBm

격자와 보간, 여러 크기의 노이즈를 합치는 원리

[원문과 공식 문서 열기](#)

Inigo Quilez · fBm / Domain Warping

노이즈 합성, 좌표를 변형해 다른 무늬를 만드는 방법

[원문과 공식 문서 열기](#)

Derivative · Noise TOP / Noise CHOP / Pattern CHOP

실제 메뉴·파라미터·데이터 형식 확인

[원문과 공식 문서 열기](#)