



RenderScript Vector Math Functions

Overview

These functions interpret the input arguments as representation of vectors in n-dimensional space.

The precision of the mathematical operations on 32 bit floats is affected by the pragmas `rs_fp_relaxed` and `rs_fp_full`. See [Mathematical Constants and Functions](#) for details.

Different precision/speed tradeoffs can be achieved by using variants of the common math functions. Functions with a name starting with

- `native_`: May have custom hardware implementations with weaker precision. Additionally, subnormal values may be flushed to zero, rounding towards zero may be used, and NaN and infinity input may not be handled correctly.
- `fast_`: May perform internal computations using 16 bit floats. Additionally, subnormal values may be flushed to zero, and rounding towards zero may be used.

Summary

Functions	
cross	Cross product of two vectors
distance	Distance between two points
dot	Dot product of two vectors
fast_distance	Approximate distance between two points
fast_length	Approximate length of a vector
fast_normalize	Approximate normalized vector
length	Length of a vector
native_distance	Approximate distance between two points
native_length	Approximate length of a vector
native_normalize	Approximately normalize a vector
normalize	Normalize a vector

Functions

cross : Cross product of two vectors

```
float3 cross(float3 left_vector, float3 right_vector);
```

```
float4 cross(float4 left_vector, float4 right_vector);
```

```
half3 cross(half3 left_vector, half3 right_vector);    Added in API level 24
```

```
half4 cross(half4 left_vector, half4 right_vector);    Added in API level 24
```

Computes the cross product of two vectors.

distance : Distance between two points

float distance(float left_vector, float right_vector);
float distance(float2 left_vector, float2 right_vector);

float distance(float3 left_vector, float3 right_vector);

float distance(float4 left_vector, float4 right_vector);

half distance(half left_vector, half right_vector); Added in [API level 24](#)

half distance(half2 left_vector, half2 right_vector); Added in [API level 24](#)

half distance(half3 left_vector, half3 right_vector); Added in [API level 24](#)

half distance(half4 left_vector, half4 right_vector); Added in [API level 24](#)

Compute the distance between two points.

See also [fast_distance\(\)](#), [native_distance\(\)](#).

dot : Dot product of two vectors

float dot(float left_vector, float right_vector);

float dot(float2 left_vector, float2 right_vector);

float dot(float3 left_vector, float3 right_vector);

float dot(float4 left_vector, float4 right_vector);

half dot(half left_vector, half right_vector); Added in [API level 24](#)

half dot(half2 left_vector, half2 right_vector); Added in [API level 24](#)

half dot(half3 left_vector, half3 right_vector); Added in [API level 24](#)

half dot(half4 left_vector, half4 right_vector); Added in [API level 24](#)

Computes the dot product of two vectors.

fast_distance : Approximate distance between two points

float fast_distance(float left_vector, float right_vector); Added in [API level 17](#)

float fast_distance(float2 left_vector, float2 right_vector); Added in [API level 17](#)

float fast_distance(float3 left_vector, float3 right_vector); Added in [API level 17](#)

float fast_distance(float4 left_vector, float4 right_vector); Added in [API level 17](#)

Computes the approximate distance between two points.

The precision is what would be expected from doing the computation using 16 bit floating point values.

See also [distance\(\)](#), [native_distance\(\)](#).

fast_length : Approximate length of a vector

float fast_length(float v); Added in [API level 17](#)

float fast_length(float2 v); Added in [API level 17](#)

float fast_length(float3 v); Added in [API level 17](#)

float fast_length(float4 v); Added in [API level 17](#)

Computes the approximate length of a vector.

The precision is what would be expected from doing the computation using 16 bit floating point values.

See also [length\(\)](#), [native_length\(\)](#).

fast_normalize : Approximate normalized vector

float fast_normalize(float v); Added in [API level 17](#)

float2 fast_normalize(float2 v); Added in [API level 17](#)

float3 fast_normalize(float3 v); Added in [API level 17](#)

[float4](#) fast_normalize([float4](#) v); Added in [API level 17](#)

Approximately normalizes a vector.

For vectors of size 1, returns -1.f for negative values, 0.f for null values, and 1.f for positive values.

The precision is what would be expected from doing the computation using 16 bit floating point values.

See also [normalize\(\)](#), [native_normalize\(\)](#).

length : Length of a vector

float length(float v);

float length([float2](#) v);

float length([float3](#) v);

float length([float4](#) v);

[half](#) length([half](#) v); Added in [API level 24](#)

[half](#) length([half2](#) v); Added in [API level 24](#)

[half](#) length([half3](#) v); Added in [API level 24](#)

[half](#) length([half4](#) v); Added in [API level 24](#)

Computes the length of a vector.

See also [fast_length\(\)](#), [native_length\(\)](#).

native_distance : Approximate distance between two points

float native_distance(float left_vector, float right_vector); Added in [API level 21](#)

float native_distance([float2](#) left_vector, [float2](#) right_vector); Added in [API level 21](#)

float native_distance([float3](#) left_vector, [float3](#) right_vector); Added in [API level 21](#)

float native_distance([float4](#) left_vector, [float4](#) right_vector); Added in [API level 21](#)

[half](#) native_distance([half](#) left_vector, [half](#) right_vector); Added in [API level 24](#)

[half](#) native_distance([half2](#) left_vector, [half2](#) right_vector); Added in [API level 24](#)

[half](#) native_distance([half3](#) left_vector, [half3](#) right_vector); Added in [API level 24](#)

[half](#) native_distance([half4](#) left_vector, [half4](#) right_vector); Added in [API level 24](#)

Computes the approximate distance between two points.

See also [distance\(\)](#), [fast_distance\(\)](#).

native_length : Approximate length of a vector

float native_length(float v); Added in [API level 21](#)

float native_length([float2](#) v); Added in [API level 21](#)

float native_length([float3](#) v); Added in [API level 21](#)

float native_length([float4](#) v); Added in [API level 21](#)

[half](#) native_length([half](#) v); Added in [API level 24](#)

[half](#) native_length([half2](#) v); Added in [API level 24](#)

[half](#) native_length([half3](#) v); Added in [API level 24](#)

[half](#) native_length([half4](#) v); Added in [API level 24](#)

Compute the approximate length of a vector.

See also [length\(\)](#), [fast_length\(\)](#).

native_normalize : Approximately normalize a vector

`float native_normalize(float v);` Added in [API level 21](#)

`float2 native_normalize(float2 v);` Added in [API level 21](#)

`float3 native_normalize(float3 v);` Added in [API level 21](#)

`float4 native_normalize(float4 v);` Added in [API level 21](#)

`half native_normalize(half v);` Added in [API level 24](#)

`half2 native_normalize(half2 v);` Added in [API level 24](#)

`half3 native_normalize(half3 v);` Added in [API level 24](#)

`half4 native_normalize(half4 v);` Added in [API level 24](#)

Approximately normalizes a vector.

See also [normalize\(\)](#), [fast_normalize\(\)](#).

normalize : Normalize a vector

`float normalize(float v);`

`float2 normalize(float2 v);`

`float3 normalize(float3 v);`

`float4 normalize(float4 v);`

`half normalize(half v);` Added in [API level 24](#)

`half2 normalize(half2 v);` Added in [API level 24](#)

`half3 normalize(half3 v);` Added in [API level 24](#)

`half4 normalize(half4 v);` Added in [API level 24](#)

Normalize a vector.

For vectors of size 1, returns -1.f for negative values, 0.f for null values, and 1.f for positive values.

See also [fast_normalize\(\)](#), [native_normalize\(\)](#).