(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第4173091号
(P4173091)

(45)発行日　平成20年10月29日(2008.10.29)　(24)登録日　平成20年8月22日(2008.8.22)

(51)Int.Cl.　F I
H05B 37/02 (2006.01)　H05B 37/02 G
H05B 37/02 B
H05B 37/02 H
H05B 37/02 L
H05B 37/02 U

請求項の数 44　(全 42 頁)

(21)出願番号　特願2003-502812 (P2003-502812)
(86)(22)出願日　平成14年6月5日(2002.6.5)
(65)公表番号　特表2004-534355 (P2004-534355A)
(43)公表日　平成16年11月11日(2004.11.11)
(86)国際出願番号　PCT/US2002/017773
(87)国際公開番号　WO2002/099780
(87)国際公開日　平成14年12月12日(2002.12.12)
審査請求日　平成17年6月3日(2005.6.3)
(31)優先権主張番号　60/296,344
(32)優先日　平成13年6月6日(2001.6.6)
(33)優先権主張国　米国(US)
(31)優先権主張番号　60/301,692
(32)優先日　平成13年6月28日(2001.6.28)
(33)優先権主張国　米国(US)

(73)特許権者　500074578
フィリップス　ソリッド－ステート　ライティング　ソリューションズ　インコーポレイテッド
アメリカ合衆国　マサチューセッツ州　０１８０３　バーリントン　バーリントン　ウッズ　ドライブ　３
(74)代理人　100102842
弁理士　葛和　清司
(72)発明者　ドーリング，ケヴィン・ジェイ
アメリカ合衆国マサチューセッツ州０１８８６，ウェストフォード，ヴィレッジ・ヴュー・ロード　２３

最終頁に続く

(54)【発明の名称】制御信号を発生するシステム及び方法

(57)【特許請求の範囲】

【請求項１】

空間を照明するための複数のアドレス可能なライトシステムを有する照明システムを制御する方法であって、

少なくともひとつの照明効果を意味する、図、写真、静的イメージ、動的イメージのうちの１つを含むグラフィック情報を供給するステップと、

空間に座標系を設定し、座標系内の位置座標を前記複数のアドレス指定可能なライトシステムに関連付けるステップと、

前記グラフィック情報のピクセルデータ位置を前記空間内の位置座標にマッピングして前記グラフィック情報を前記ライトシステムを制御する信号に変換し、前記グラフィック情報が意味する照明効果を発生するように前記空間を照明するステップとを備える方法。

【請求項２】

前記ライトシステムがネットワーク化されたライトシステムであり、前記照明制御信号がアドレス指定された情報のパケットにパッケージされる請求項１に記載の方法。

【請求項３】

前記グラフィック情報がコンピュータのスクリーン上に表示される請求項１に記載の方法。

【請求項４】

グラフィック情報を供給する前記ステップが、コンピュータを使用して前記グラフィック情報を発生するステップを含む請求項１に記載の方法。

【請求項５】
前記グラフィック情報が、ビットマップ及びベクトル座標のうちの少なくとも１つを使用して発生される請求項４に記載の方法。
【請求項６】
前記グラフィック情報は３D空間内に描画されたものである請求項４に記載の方法。
【請求項７】
前記グラフィック情報が関数によって発生される請求項４に記載の方法。
【請求項８】
前記関数が、室内での光の回転、室内での光のボールの跳ね返り、及び室内の音の跳ね返りのうちの少なくともひとつを含むイメージを表す請求項７に記載の方法。
【請求項９】
前記関数がランダムに発生される効果を表す請求項７に記載の方法。
【請求項１０】
前記関数が前記システムへの入力に関係する請求項７に記載の方法。
【請求項１１】
前記入力が、情報、ファイル、音楽、信号、データ・ストリーム、ボイス・ストリーム、無線データ・ストリーム、及び感知された状態のうちの少なくとも１つを含む請求項１０に記載の方法。
【請求項１２】
前記グラフィック情報は、コンピュータのスクリーン上に表示せずに前記ライトシステムを制御するように変換される請求項１に記載の方法。
【請求項１３】
前記制御信号が、色、光度、範囲、前記ライトシステムを使用して生成された効果の伝搬速度のうちの少なくとも１つを制御する信号を含む請求項１に記載の方法。
【請求項１４】
前記制御信号が前記ライトシステムを制御して前記空間を照明し、事象をシミュレートする効果を発生させる、請求項13に記載の方法。
【請求項１５】
前記事象が、爆発、照明の当たり、ヘッドライト、室内の列車通過、室内を通る弾丸発射、室内を通る光の移動、室内の日の出及びその他の事象の少なくとも１つを含む、請求項１４に記載の方法。
【請求項１６】
信号を使用して、前記ライトシステムを制御し、指定された時間に照明を行う請求項１に記載の方法。
【請求項１７】
グラフィック情報のピクセルデータ位置を前記空間内の位置座標にマッピングするステップが、グラフィカル・ユーザ・インターフェースを使用して行われる請求項１に記載の方法。
【請求項１８】
マッピングが、１つ又は複数のピクセル、ピクセルの群、ベクトル座標情報、波動関数を空間内のライトシステムの位置にマッピングすることを含む請求項17に記載の方法。
【請求項１９】
前記空間が、部屋、廊下、ホール、建物、表示装置、ブース、劇場、小売店、店舗、棚、対象物、及び製品のうちの少なくともひとつを含む請求項18に記載の方法。
【請求項２０】
前記ライトシステムによって照明される表面の表現に対する位置マップを発生するステップを更に備える請求項１に記載の方法。
【請求項２１】
前記位置マップの発生が前記空間内の光のを撮像装置で補足することにより自動的に行われる請求項２０に記載の方法。

【請求項２２】
前記ライトシステムを制御するための制御信号を送信する前に効果をスクリーン上に視覚化するための前記スクリーンを設けるステップを更に備える請求項１に記載の方法。
【請求項２３】
空間を照明するための複数のアドレス可能なライトシステムを有する照明システムを制御するシステムであって、
図、写真、静的イメージ、動的イメージのうちの少なくとも１つを含むグラフィック情報を供給するように適合されているコンピュータ・アプリケーションと、
前記空間内の座標系における位置座標を、前記複数のアドレス指定可能なライトシステムの各々に対応付けるように適合されている関連付けシステムとを含み、
前記グラフィック情報のピクセルデータ位置を、前記座標系における位置座標にマッピングするように適合されているマッピングシステムと、
前記グラフィック情報を、前記ライトシステムを制御するための制御信号に変換して、前記グラフィック情報が意味する照明効果となるように前記空間を照明するように適合されている変換器と
を備えることを特徴とするシステム。
【請求項２４】
前記ライトシステムが、ネットワーク化されたライトシステムであり、前記制御信号が、アドレス指定された情報のパケットにパッケージされる請求項２３に記載のシステム。
【請求項２５】
前記グラフィック情報がコンピュータのスクリーン上に表示される請求項２３に記載のシステム。
【請求項２６】
前記コンピュータ・アプリケーションが前記グラフィック情報を発生するように適合されている請求項２３に記載のシステム。
【請求項２７】
前記グラフィック情報が３D空間で発生される請求項２６に記載のシステム。
【請求項２８】
前記グラフィック情報が３D空間内に描画される請求項２６に記載のシステム。
【請求項２９】
前記グラフィック情報が関数によって発生される請求項２６に記載のシステム。
【請求項３０】
前記関数が、室内での光の回転、室内での光のボールの跳ね返り、及び室内の音の跳ね返りのうちの少なくとも１つのイメージを表す請求項２９に記載のシステム。
【請求項３１】
前記関数がランダムに発生される効果を表す請求項２９に記載のシステム。
【請求項３２】
前記関数が前記システムへの入力に関係する請求項２９に記載のシステム。
【請求項３３】
前記入力が、情報、ファイル、音楽、信号、データ・ストリーム、ボイス・ストリーム、無線データ・ストリーム、及び感知された状態のうちの少なくとも１つを含む請求項３２に記載のシステム。
【請求項３４】
前記コンピュータ・アプリケーションが、コンピュータのスクリーンにグラフィック情報を表示せずに、前記グラフィック情報を供給するように適合されている請求項２３に記載のシステム。
【請求項３５】
前記制御信号が、色、光度、範囲、及び前記ライトシステムを使用して生成された効果の伝搬速度のうちの少なくとも１つを制御する信号を含む請求項２３に記載のシステム。
【請求項３６】

前記制御信号が前記ライトシステムを制御して前記空間を照明し、事象をシミュレートする効果を発生させる、請求項３５に記載のシステム。

【請求項３７】

前記事象が、爆発、照明の当たり、ヘッドライト、室内の列車通過、室内を通る弾丸発射、室内を通る光の移動、室内の日の出、及びその他の事象からなるグループから選択される請求項３６に記載のシステム。

【請求項３８】

前記制御信号を使用して、前記ライトシステムを制御し、指定された時間に照明を行う請求項２３に記載のシステム。

【請求項３９】

前記関連付けシステムが、グラフィカル・ユーザ・インターフェースに関連付けられ、前記グラフィカル・ユーザ・インターフェースを使用して、前記複数のアドレス指定可能なライトシステムを前記環境内の場所に関連付ける請求項２３に記載のシステム。

【請求項４０】

前記グラフィカル・ユーザ・インターフェースが空間の表現を含む請求項３９に記載のシステム。

【請求項４１】

前記空間が、部屋、廊下、ホール、建物、表示装置、ブース、劇場、小売店、店舗、棚、対象物、及び製品からなるグループから選択される請求項４０に記載のシステム。

【請求項４２】

前記ライトシステムによって照明される表面の表現に対する位置マップを発生するように適合されている位置マップ発生器を更に備える請求項２３に記載のシステム。

【請求項４３】

前記位置マップの発生が前記空間内の光のを撮像装置で補足することにより自動的に行われる請求項４２に記載のシステム。

【請求項４４】

前記ライトシステムを制御するための制御信号を送信する前に効果をスクリーン上に視覚化するための前記スクリーンを更に備える請求項２３に記載のシステム。

【発明の詳細な説明】

【技術分野】

【０００１】

［関連出願の相互参照］

本発明は、２００１年6月6日に出願された「制御信号を発生するシステム及び方法（Ｓｙｓｔｅｍｓ ａｎｄ Ｍｅｔｈｏｄｓ ｏｆ Ｇｅｎｅｒａｔｉｎｇ Ｃｏｎｔｒｏｌ Ｓｉｇｎａｌｓ）」という表題の米国特許仮出願第６０／２９６３４４号明細書、２００１年6月２８日に出願された「ＬＥＤ照明システムをネットワーク化するシステム及び方法（Ｓｙｓｔｅｍｓ ａｎｄ Ｍｅｔｈｏｄｓ ｆｏｒ Ｎｅｔｗｏｒｋｉｎｇ ＬＥＤ Ｌｉｇｈｔｉｎｇ Ｓｙｓｔｅｍｓ）」という表題の米国特許仮出願第６０／３０１６９２号明細書、２００１年１０月１２日に出願された「ＬＥＤ照明システムをネットワーク化するシステム及び方法（Ｓｙｓｔｅｍｓ ａｎｄ Ｍｅｔｈｏｄｓ ｆｏｒ Ｎｅｔｗｏｒｋｉｎｇ ＬＥＤ Ｌｉｇｈｔｉｎｇ Ｓｙｓｔｅｍｓ）」という表題の米国特許仮出願第６０／３２８８６７号明細書、および２００１年１０月３０日に出願された「ＬＥＤ照明のためのシステム及び方法（Ｓｙｓｔｅｍｓ ａｎｄ Ｍｅｔｈｏｄｓ ｆｏｒ ＬＥＤ Ｌｉｇｈｔｉｎｇ）」という表題の米国特許仮出願第６０／３４１４７６号明細書の特権を米国特許法第１１９条（ｅ）に基づき主張する。

【０００２】

本出願は、さらに、１９９７年8月２６日に出願され、現在は米国特許第６０１６０３８号となっている、米国特許出願第０８／９２０１５６号の継続出願である、１９９９年１０月２２日に出願され、現在は米国特許第６１５０７７４号となっている、米国特許出願第０９／４２５７７０号の継続出願である、２０００年9月２５日に出願された「多色

ＬＥＤ照明方法及び装置（Ｍｕｌｔｉｃｏｌｏｒｅｄ ＬＥＤ Ｌｉｇｈｔｉｎｇ Ｍｅｔｈｏｄ ａｎｄ Ａｐｐａｒａｔｕｓ）」という表題の米国特許非仮出願第０９／６６９１２１号の継続出願である、２００１年１０月４日に出願された「多色ＬＥＤ照明方法及び装置（Ｍｕｌｔｉｃｏｌｏｒｅｄ ＬＥＤ Ｌｉｇｈｔｉｎｇ Ｍｅｔｈｏｄ ａｎｄ Ａｐｐａｒａｔｕｓ）」という表題の米国特許非仮出願第０９／９７１３６７号の一部継続出願（ＣＩＰ）として米国特許法第１２０条に基づき特権を主張する。

【０００３】

本出願はさらに、２００１年5月３０日に出願された「ネットワーク化された照明システムにおける装置を制御する方法及び装置（Ｍｅｔｈｏｄｓ ａｎｄ Ａｐｐａｒａｔｕｓ ｆｏｒ Ｃｏｎｔｒｏｌｌｉｎｇ Ｄｅｖｉｃｅｓ ｉｎ ａ Ｎｅｔｗｏｒｋｅｄ Ｌｉｇｈｔｉｎｇ Ｓｙｓｔｅｍ）」という表題の米国特許非仮出願第０９／８７０１９３号、１９９８年１２月１７日に出願された「スマート白熱電球（Ｓｍａｒｔ Ｌｉｇｈｔ Ｂｕｌｂ）」という表題の米国特許非仮出願第０９／２１５６２４号、１９９８年１２月１７日に出願された「センサ応答型照明システム及び方法（Ｓｙｓｔｅｍｓ ａｎｄ Ｍｅｔｈｏｄｓ ｆｏｒ Ｓｅｎｓｏｒ－Ｒｅｓｐｏｎｓｉｖｅ Ｉｌｌｕｍｉｎａｔｉｏｎ）」という表題の米国特許非仮出願第０９／２１３６０７号、１９９８年１２月１７日に出願された「精密照明（Ｐｒｅｃｉｓｉｏｎ Ｉｌｌｕｍｉｎａｔｉｏｎ）」という表題の米国特許非仮出願第０９／２１３１８９号、１９９８年１２月１７日に出願された「動的照明（Ｋｉｎｅｔｉｃ Ｉｌｌｕｍｉｎａｔｉｏｎ）」という表題の米国特許非仮出願第０９／２１３５８１号、１９９８年１２月１７日に出願された「データ配送トラック（Ｄａｔａ Ｄｅｌｉｖｅｒｙ Ｔｒａｃｋ）」という表題の米国特許非仮出願第０９／２１３５４０号、１９９９年6月１５日に出願された「拡散照明システム及び方法（Ｄｉｆｆｕｓｅ Ｉｌｌｕｍｉｎａｔｉｏｎ Ｓｙｓｔｅｍｓ ａｎｄ Ｍｅｔｈｏｄｓ）」という表題の米国特許非仮出願第０９／３３３７３９号、１９９８年１２月１７日に出願され、現在は米国特許第６１６６４９６号となっている、米国特許出願第０９／２１３５４８号の継続出願である、２００１年3月２２日に出願された「照明娯楽システム（Ｌｉｇｈｔｉｎｇ Ｅｎｔｅｒｔａｉｎｍｅｎｔ Ｓｙｓｔｅｍ）」という表題の米国特許非仮出願第０９／８１５４１８号、２００１年１０月２３日に出願された「ディジタル娯楽システム及び方法（Ｓｙｓｔｅｍｓ ａｎｄ Ｍｅｔｈｏｄｓ ｆｏｒ Ｄｉｇｉｔａｌ Ｅｎｔｅｒｔａｉｎｍｅｎｔ）」という表題の米国特許非仮出願第１０／０４５６０４号、２００１年１１月２０日に出願された「自動車情報システム（Ａｕｔｏｍｏｔｉｖｅ Ｉｎｆｏｒｍａｔｉｏｎ Ｓｙｓｔｅｍｓ）」という表題の米国特許非仮出願第０９／９８９０９５号、２００１年１１月２０日に出願された「パッケージ型情報システム（Ｐａｃｋａｇｅｄ Ｉｎｆｏｒｍａｔｉｏｎ Ｓｙｓｔｅｍｓ）」という表題の米国特許非仮出願第０９／９８９７４７号、２００１年１１月２０日に出願された「情報システム（Ｉｎｆｏｒｍａｔｉｏｎ Ｓｙｓｔｅｍｓ）」という表題の米国特許非仮出願第０９／９８９６７７号の一部継続出願（ＣＩＰ）として米国特許法第１２０条に基づき特権を主張する。

【０００４】

本出願は、さらに、上記の米国特許非仮出願のうち少なくとも１つが、同様に、１９９７年１２月１７日に出願された「ディジタル制御された発光ダイオード・システム及び方法（Ｄｉｇｉｔａｌｌｙ Ｃｏｎｔｒｏｌｌｅｄ Ｌｉｇｈｔ Ｅｍｉｔｔｉｎｇ Ｄｉｏｄｅｓ Ｓｙｓｔｅｍｓ ａｎｄ Ｍｅｔｈｏｄｓ）」という表題の米国特許仮出願第６０／０７１２８１号、１９９７年１２月２４日に出願された「多色インテリジェント照明（Ｍｕｌｔｉ－Ｃｏｌｏｒ Ｉｎｔｅｌｌｉｇｅｎｔ Ｌｉｇｈｔｉｎｇ）」という表題の米国特許仮出願第６０／０６８７９２号、１９９８年3月２０日に出願された「ディジタル照明システム（Ｄｉｇｉｔａｌ Ｌｉｇｈｔｉｎｇ Ｓｙｓｔｅｍｓ）」という表題の米国特許仮出願第６０／０７８８６１号、１９９８年3月２５日に出願された「制御された照明のためのシステム及び方法（Ｓｙｓｔｅｍ ａｎｄ Ｍｅｔｈｏｄ ｆｏｒ Ｃｏｎｔｒｏｌｌｅｄ Ｉｌｌｕｍｉｎａｔｉｏｎ）」という表題の米国特許仮出願第６

０／０７９２８５号、１９９８年６月２６日に出願された「多重同時高速パルス幅変調信号のソフトウエア駆動された発生方法（Ｍｅｔｈｏｄｓ ｆｏｒ Ｓｏｆｔｗａｒｅ Ｄｒｉｖｅｎ Ｇｅｎｅｒａｔｉｏｎ ｏｆ Ｍｕｌｔｉｐｌｅ Ｓｉｍｕｌｔａｎｅｏｕｓ Ｈｉｇｈ Ｓｐｅｅｄ Ｐｕｌｓｅ Ｗｉｄｔｈ Ｍｏｄｕｌａｔｅｄ Ｓｉｇｎａｌｓ）」という表題の米国特許仮出願第６０／０９０９２０号、２００１年３月２２日に出願された「ディジタル娯楽システム及び方法（Ｓｙｓｔｅｍｓ ａｎｄ Ｍｅｔｈｏｄｓ ｆｏｒ Ｄｉｇｉｔａｌ Ｅｎｔｅｒｔａｉｎｍｅｎｔ）」という表題の米国特許仮出願第６０／２７７９１１号、２０００年１０月２３日に出願された「ディジタル娯楽システム（Ｓｙｓｔｅｍｓ ａｎｄ Ｍｅｔｈｏｄｓ ｆｏｒ Ｄｉｇｉｔａｌ Ｅｎｔｅｒｔａｉｎｍｅｎｔ）」という表題の米国特許仮出願第６０／２４２４８４号、２０００年１１月２０日に出願された「インテリジェント・インディケータ（Ｉｎｔｅｌｌｉｇｅｎｔ Ｉｎｄｉｃａｔｏｒｓ）」という表題の米国特許仮出願第６０／２５２００４号、２００１年１月１６日に出願された「色変化ＬＣＤスクリーン（Ｃｏｌｏｒ Ｃｈａｎｇｉｎｇ ＬＣＤ Ｓｃｒｅｅｎｓ）」という表題の米国特許仮出願第６０／２６２０２２号、２００１年１月１７日に出願された「情報システム（Ｉｎｆｏｒｍａｔｉｏｎ Ｓｙｓｔｅｍｓ）」という表題の米国特許仮出願第６０／２６２１５３号、２００１年２月１３日に出願された「車両のためのＬＥＤベース型照明システム（ＬＥＤ Ｂａｓｅｄ Ｌｉｇｈｔｉｎｇ Ｓｙｓｔｅｍｓ ｆｏｒ Ｖｅｈｉｃａｌｓ）」という表題の米国特許仮出願第６０／２６８２５９号、および２００１年６月６日に出願された「情報を表示するシステム及び方法（Ｓｙｓｔｅｍｓ ａｎｄ Ｍｅｔｈｏｄｓ ｆｏｒ Ｄｉｓｐｌａｙｉｎｇ Ｉｎｆｏｒｍａｔｉｏｎ）」という表題の米国特許仮出願第６０／２９６２１９号のうちの少なくとも１つの特権が与えられるため、これらの特許仮出願のそれぞれの特権も米国特許法第１２０条に基づき主張する。

前記の出願はそれぞれ、参照により本明細書に組み込まれている。

【０００５】

［発明の背景］

出現するさまざまな照明条件に対応するため、ネットワーク接続型照明制御が次第に普及してきている。Ｃｏｌｏｒ Ｋｉｎｅｔｉｃｓ Ｉｎｃｏｒｐｏｒａｔｅｄでは、ネットワーク化された照明システムだけでなくコントローラおよびライトショーのオーサリング・ツールを各種揃えている。一般に、照明システム用制御信号が、発生され、ネットワーク経由で複数の照明システムに通信される。いくつかの照明システムを照明ネットワーク構成とし、各照明デバイスに関係する情報をネットワーク経由で通信することができる。各照明デバイスまたはシステムは、特定のアドレスを宛て先とする情報のみを読み出し、反応するように一意的な識別子またはアドレスを有することができる。

【０００６】

ネットワーク化された照明制御信号の発生に使用される方法はいくつかある。制御信号発生ツールは、ライトショーおよびシーケンスをオーサリングできるグラフィカル・ユーザ・インターフェースを提供することができる。ユーザは、一連のアドレス指定された照明システムをセットアップし、次いで、個別にアドレス指定された照明システムに指向される照明制御信号を生成することができる。このようなオーサリング・システムを使用することにより、照明システム間または照明システムのグループ内で調和した効果を発生することができる。オーサリング・システムがないとプログラミングが難しいが特に人気のある照明効果として、廊下の複数の照明具の色循環を時間差をつけながら連続的にオンし虹色が流れていくようにする効果がある。

【０００７】

調和した照明効果を発生するために、ユーザは従来、照明システムのある場所を知っているだけでなく、各照明システムの特定のアドレスも知っていなければならない。それでも、一列になっている領域や照明システムのグループ内以外の範囲で移動するように設計されている照明効果をプログラミングすることは困難である。ユーザがある範囲内で所望の効果に基づき照明制御信号を発生し、通信することができるシステムを実現できれば有

益であろう。

【０００８】

［発明の概要］

本明細書では、照明システム用の制御信号を発生する方法およびシステムを提供する。この方法およびシステムは、複数の照明システムの位置をマッピングするためのライト管理機構を設けるステップと、複数の照明システムの位置をマッピングするマップ・ファイルを発生するステップと、コンピュータ・アプリケーションを使用して効果を発生するステップと、照明システムの特性にコンピュータ・アプリケーションのためのコードを関連付けるステップと、照明システムを制御するための照明制御信号を発生するステップとを含む。

【０００９】

本明細書では、照明システムを制御する方法およびシステムを提供する。この方法およびシステムは、グラフィック情報を供給するステップと、複数のアドレス指定可能な照明システムに環境内での場所を関連付けるステップと、グラフィック情報を、制御システムを制御することができる制御信号に変換して、このグラフィック情報に応じて環境を照明するステップを含み得る。

【００１０】

本明細書では、照明システムを制御する方法およびシステムを提供する。この方法およびシステムは、グラフィックを生成するための一組の情報にアクセスするステップと、複数のアドレス指定可能な照明システムに環境内での場所を関連付けるステップと、アルゴリズムをグラフィック情報に適用して、当該グラフィック情報を、照明システムを制御することができる制御信号に変換して、このグラフィック情報に応じて環境内に照明効果を生成するステップとを含み得る。

【００１１】

本発明では、複数の照明システムに環境内の位置を自動的に関連付けるための方法およびシステムを提供する。この方法およびシステムは、照明システムのイメージを捕捉するための像形成装置にアクセスするステップと、所定の順序でオンするよう複数の照明システムのそれぞれを指令するステップと、複数の照明システムのそれぞれについて「オン」時間中にイメージを捕捉するステップと、イメージ内の照明システムの位置に基づき環境内の照明システムの位置を計算するステップを含み得る。

【００１２】

本発明では、環境内で照明効果を発生する方法およびシステムを提供する。この方法およびシステムは、非照明システムを使用してイメージを発生するステップと、複数の照明システムに環境内での位置を関連付けるステップと、照明システムと位置の上記の関連付けを使用して、イメージを照明システム用の制御信号に変換するステップとを備え、上記照明システムがそのイメージに対応する効果を発生する。

【００１３】

本発明では、照明システム用の制御信号を発生する方法およびシステムを提供する。この方法およびシステムは、複数の照明システムの位置をマッピングするためのライト管理機構を設けるステップと、ライト管理機構を使用して、複数の照明システムの位置をマッピングするマップ・ファイルを発生するステップと、アニメーション機構を使用して、複数のグラフィックス・ファイルを発生するステップと、マップ・ファイル内の照明システムの位置にグラフィックス・ファイル内のデータを関連付けるステップと、グラフィックス・ファイルと関連して照明システムを制御する照明制御信号を発生するステップとを含み得る。

【００１４】

本発明では、照明システムを制御する方法およびシステムを提供する。この方法およびシステムは、環境内の複数の照明システム用の照明制御信号を取得するステップと、コンピュータからグラフィックス信号を取得するステップと、グラフィックス信号の内容に応答して照明制御信号を修正するステップとを含み得る。

【００１５】
本発明により、従来技術にかかわる問題の多くを解消することができる。本発明の一実施形態は、制御信号を発生するシステムである。このシステムを使用すると、ユーザはイメージ、イメージの表現、アルゴリズム、またはその他の効果情報を発生することができる。効果情報は、次いで、照明制御信号に変換されて、保存されるか、またはネットワーク化された照明システムに通信することができる。本発明の一実施形態では、ある空間または範囲内に効果が発生するように制御信号のオーサリング、発生、および通信を可能にすることができる。
【００１６】
一実施形態では、照明システム、照明ネットワーク、ライト、ＬＥＤ、ＬＥＤ照明システム、オーディオ・システム、サラウンド・サウンド・システム、霧発生装置、降雨装置、電気機械システム、またはその他のシステムを制御することができる制御信号を発生することができる。
【００１７】
本発明の原理によるシステムは、イメージ情報を発生し、そのイメージ情報を、ネットワーク化された照明システムを制御することができる制御信号に変換することができる。一実施形態では、範囲または空間内に配置される複数のアドレス指定可能照明システムを識別するための構成情報を発生することができる。一実施形態では、照明システムに照明が当たる表面を関連付ける構成情報を発生することができる。一実施形態では、制御信号は、複数のアドレス指定された照明システムを含む照明ネットワークに通信することができる。一実施形態では、音響または他の効果を照明制御信号により調整することができる。
【００１８】
図では、本発明のいくつかの例示的な実施形態が説明されており、類似の参照番号は類似の構成要素を指している。図に示されている実施形態は、本発明の例示として理解すべきであって、いかなる形でも制限しているものと解釈すべきではない。
【００１９】
［好適な実施形態の詳細な説明］
以下の説明は、本発明のいくつかの例示的な図に示した実施形態に関するものである。当業者であれば本発明について多くの変形を思いつくことができるが、そのような変形および改良は本開示の範囲内にある。そこで、本発明の範囲はいかなる点でも以下の開示によって制限されないものとする。
【００２０】
本発明の一実施形態は、制御信号を発生するシステムと方法に関する。制御信号を使用することにより、照明システム、照明ネットワーク、ライト、ＬＥＤ、ＬＥＤ照明システム、オーディオ・システム、サラウンド・サウンド・システム、霧発生装置、降雨装置、電気機械システム、またはその他のシステムを制御することができる。米国特許第６０１６０３８号、第６１５０７７４号、および第６１６６４９６号で説明されているような照明システムは、制御信号が使用されるいくつかの異なる種類の照明システムである。
【００２１】
本発明の全体を理解できるようにするため、ＬＥＤベースのシステムを含むプログラム可能なライトおよび照明システム用のさまざまなアプリケーションを含む、いくつかの例示的な実施形態について説明することにする。しかし、当業者であれば、本明細書で説明している方法およびシステムは、プログラム可能な照明があることが望ましい他の環境にもうまく適合させられること、および本明細書で説明している実施形態がＬＥＤベースでない照明にも適合できることを理解するであろう。また、当業者であれば、以下で説明する実施形態は、照明制御システム用のオーサリング・ツールである必要のない種類のコンピュータソフトウェアだけでなく、さまざまな他の種類のコンピュータ・アプリケーションとともに使用することもできることを理解するであろう。さらに、ユーザは、コンピュータを操作する必要はないが、そのユーザに情報を供給するソフトウェアアプリケーショ

ンを実行できる種類のコンピューティング・デバイスであればそのようなデバイスを操作することができる。

【００２２】

いくつかのコンピュータ・アプリケーションでは、通常、何らかの種類の仮想環境を表示する表示スクリーン（コンピューティング・デバイス、テレビジョン・スクリーン、ラップトップ・スクリーン、ハンドヘルド・スクリーン、ゲームボーイ・スクリーン、コンピュータのモニタ、平面型表示装置、ＬＣＤ表示装置、ＰＤＡスクリーン、またはその他の表示装置）がある。通常、ユーザは、表示スクリーンを囲む現実世界の環境内にいる。本発明は、とりわけ、仮想環境内でコンピュータ・アプリケーションを使用して、現実世界の環境内に置かれている照明システムなどのシステム用の制御信号を発生することに関する。

【００２３】

図１では、本明細書で説明している発明の一実施形態において、環境１００は１つまたは複数のライト・システム１０２を備える。本明細書で使用しているように、「照明システム」は、ＬＥＤシステムだけでなく、白熱電球などの白熱光源、火炎などのピロルミネセンス光源、ガスマントルおよびカーボンアーク放射源などのキャンドル・ルミネセンス光源、気体放電などの光輝性光源、蛍光光源、燐光光源、レーザー、エレクトロルミネセント・ランプなどのエレクトロルミネセンス光源、発光ダイオード、および電子飽和を使用した陰極線ルミネセンス光源、さらにガルバノルミネセンス光源、結晶ルミネセンス光源、キネルミネセンス光源、熱ルミネセンス光源、トリボルミネセンス光源、音ルミネセンス光源、放射線ルミネセンス光源などの種々のルミネセンス光源を含む、すべての照明システムを備える状況が適切である場所において理解すべきである。さらに、ライト・システム１０２は原色などの色を生成することができる発光性ポリマーも含む。好ましい一実施形態では、ライト・システム１０２はＬＥＤベースの照明システムである。好ましい一実施形態のライト・システム１０２では、２色の光を混合することができ、光の色は赤色、緑色、青色、白色、琥珀色、またはその他の色である。一実施形態では、光の色は異なる色の白色光、つまり、色温度が異なる白色光とすることができる。

【００２４】

本明細書では「ＬＥＤ」という用語を使用しているが、これは、電気信号を受信し、その信号に応答してある色の光を生成することができるシステムを意味する。そのため、「ＬＥＤ」という用語は、あらゆる種類の発光ダイオード、例えば、発光ポリマー、電流に応答して光を発する半導体ダイ、有機ＬＥＤ、電子発光ストリップ、および他のそのようなシステムを含むことを理解されなければならない。一実施形態では、「ＬＥＤ」は、個別に制御される複数の半導体ダイを持つ単一発光ダイオードを意味することができる。ここでもまた、「ＬＥＤ」という用語はＬＥＤのパッケージ・タイプを制約するものではないことを理解されたい。「ＬＥＤ」という用語は、パッケージ化されたＬＥＤ、パッケージ化されていないＬＥＤ、表面実装ＬＥＤ、チップオンボードＬＥＤ、および他のすべての構成のＬＥＤを含む。「ＬＥＤ」という用語はさらに、パッケージングされた、または蛍光体と関連付けられているＬＥＤも含み、蛍光体はＬＥＤからのエネルギーを異なる波長に変換することができる。ＬＥＤシステムは、照明源の一種である。

【００２５】

「照明する（ｉｌｌｕｍｉｎａｔｅ）」という用語は、照明源によるある周波数の放射線の生成を意味するものと理解すべきである。「光」および「色」という用語は、あるスペクトル範囲内の任意の周波数の放射線を意味することが適切である状況において理解すべきである、つまり、本明細書で使用しているように「光」の「色」は、白色光を含む可視スペクトルの周波数または周波数の組み合わせだけでなく、スペクトルの赤外線および紫外線領域、および電磁スペクトルの他の領域の周波数をも含むものと理解すべきである。

【００２６】

図２は、照明システム２００の一実施形態を説明するブロック図である。プロセッサ２

０４は、いくつかのライト２０８と関連付けられている。プロセッサは、制御信号をライト２０８に送信する。このようなシステムでは、任意選択により、プロセッサとライト２０８の間に１つまたは複数のコントローラ、トランジスタ又は類似のもののような１つまたは複数の中間構成要素を配置することができる。

【００２７】

本明細書で使用しているように、プロセッサという用語は、電子信号を処理する任意のシステムを指す。プロセッサとしては、マイクロプロセッサ、マイクロコントローラ、プログラム可能ディジタル信号プロセッサ、その他のプログラム可能デバイス、コントローラ、アドレス指定可能コントローラ、マイクロプロセッサ、マイクロコントローラ、アドレス指定可能マイクロプロセッサ、コンピュータ、プログラム可能プロセッサ、プログラム可能コントローラ、専用プロセッサ、専用コントローラ、集積回路、制御回路、またはその他のプロセッサがある。プロセッサには、さらには、あるいは、その代わりとして、特定用途向け集積回路、プログラム可能ゲート・アレイ、プログラム可能アレイ・ロジック、プログラム可能論理デバイス、ディジタル信号プロセッサ、アナログ／ディジタル変換器、ディジタル／アナログ変換器、または電子信号を処理するように構成されていることができるその他のデバイスがある。さらに、プロセッサは、抵抗器、コンデンサ、インダクタ、トランジスタ、演算増幅器などを含む受動または能動アナログ構成要素などの個別の回路だけでなく、論理構成要素、シフト・レジスタ、ラッチ、またはディジタル機能を実現するため個別にパッケージングされたチップまたはその他の構成要素などの個別のディジタル構成要素も備えることができる。上記の回路および構成要素の任意の組み合わせを、パッケージングが個別であろうと、チップであろうと、チップセットであろうと、またはダイであろうと、本明細書で説明しているようなプロセッサとして使用するように適切に構成することができる。さらに、プロセッサという用語は、パーソナル・コンピュータ、ネットワーク・サーバなどの統合化されたシステム、あるいは自律的にまたは本明細書で説明しているような電子信号を処理するためコマンドに応答して動作することが可能な他のシステムにも適用できることは理解されるであろう。プロセッサに上述のマイクロプロセッサまたはマイクロコントローラなどのプログラム可能なデバイスが含まれる場合、プロセッサはさらに、プログラム可能なデバイスの動作を制御するコンピュータ実行可能コードを備えることができる。一実施形態では、プロセッサ２０４は、Ｍｉｃｒｏｃｈｉｐ　ＰＩＣプロセッサ１２Ｃ６７２およびライト２０８は赤色ＬＥＤ、緑色ＬＥＤ、および青色ＬＥＤなどのＬＥＤである。

【００２８】

プロセッサ２０４は、任意選択により、パルス幅変調器、パルス振幅変調器、パルス変位変調器、抵抗器ラダー回路、電流源、電圧源、電圧ラダー回路、スイッチ、トランジスタ、電圧コントローラ、またはその他のコントローラなどのさまざまな他の構成要素および制御構成要素（図には示されていない）を含めるか、あるいはそれらと関連して使用することができる。制御構成要素およびプロセッサ２０４は、ライト２０８を通じて電流、電圧、および／または電力を制御することができる。

【００２９】

一実施形態では、異なるスペクトル出力を持ついくつかのＬＥＤをライト２０８として使用することができる。色のそれぞれを個別の制御チャネルを通じて駆動することができる。プロセッサ２０４およびコントローラは、１つのデバイスに組み込むことができる。このデバイスは、ストリング内のいくつかのＬＥＤを駆動する機能に電力を供給したり、あるいは直接１つまたは少数のＬＥＤにのみ対応できるようにすることも可能である。プロセッサ２０４およびコントローラは、別々のデバイスであってもよい。ＬＥＤを独立に制御することにより、色を混合して照明効果を生み出すことができる。

【００３０】

一実施形態では、メモリ２１０も備えることができる。メモリ２１０は、制御信号と関連付けられたアルゴリズム、テーブル、または値を格納することができる。メモリ２１０は、プロセッサ２０４、その他の構成要素、およびライト２０８を制御するプログラムを

格納することができる。メモリ２１０には、メモリ、読み取り専用メモリ、プログラム可能メモリ、プログラム可能読み取り専用メモリ、電子的消去可能プログラム可能読み取り専用メモリ、ランダム・アクセス・メモリ、ダイナミック・ランダム・アクセス・メモリ、ダブル・データ・レート・ランダム・アクセス・メモリ、Ｒａｍｂｕｓダイレクト・ランダム・アクセス・メモリ、フラッシュメモリ、またはその他の揮発性もしくは不揮発性メモリを使用することができ、プログラム命令、プログラムデータ、アドレス情報、およびプログラム出力またはその他の中間または最終結果を格納することができる。

【００３１】

例えば、プログラムにより、いくつかの異なる発光色のライト２０８を動作させるための制御信号を格納することができる。ユーザ・インターフェース２０２も、任意選択により、プロセッサ２０４と関連付けることができる。ユーザ・インターフェース２０２を使用することで、メモリからプログラムを選択し、メモリからプログラムを修正し、メモリからプログラム・パラメータの修正し、外部信号を選択し、又は他のユーザ・インターフェースによる解決手段を与えることができる。色混合およびパルス幅変調制御のいくつかの方法が、参照により開示全体が本明細書に組み込まれている、「多色ＬＥＤ照明方法及びシステム（Ｍｕｌｔｉｃｏｌｏｒｅｄ　ＬＥＤ　Ｌｉｇｈｔｉｎｇ　Ｍｅｔｈｏｄ　ａｎｄ　Ａｐｐａｒａｔｕｓ）」という表題の米国特許第６０１６０３８号で開示されている。プロセッサ２０４はさらに、アドレス指定可能であって、このプロセッサにアドレス指定されたプログラミング信号を受信することもできる。例えば、プロセッサ２０４は、複数の類似のプロセッサまたはその他のデバイスのためのデータ要素を含むデータ（または照明制御信号）のストリームを受信し、そしてプロセッサ２０４は、そのプロセッサにアドレス指定されている適切なデータ要素を当該ストリームから抽出することができる。一実施形態では、ユーザ・インターフェースは、以下で詳しく説明するような照明制御信号を発生するためのオーサリング・システムを備えることができる。

【００３２】

ＬＥＤの制御は、著しく発展した。ＬＥＤ制御の分野における米国特許としては、米国特許第６０１６０３８号、第６１５０７７４号、および第６１６６４９６号がある。「照明条件を発生し且つ変調するシステム及び方法（Ｓｙｓｔｅｍｓ　ａｎｄ　Ｍｅｔｈｏｄｓ　ｆｏｒ　Ｇｅｎｅｒａｔｉｎｇ　ａｎｄ　Ｍｏｄｕｌａｔｉｎｇ　Ｉｌｌｕｍｉｎａｔｉｏｎ　Ｃｏｎｄｉｔｉｏｎｓ）」の米国特許出願第０９／７１６８１９号でも、とりわけシステムと制御装置類について説明している。これらすべての文書の開示全体が、参照により本明細書に組み込まれている。

【００３３】

本発明の複数の施形態では、照明システムを使用して環境を照明することができる。このような環境１００の１つが図１に示されている。この環境では、少なくとも１つのライト・システム１０２を取り付けており、好ましい一実施形態では、複数のライト・システム１０２を装着することができる。ライト・システム１０２は、図２に関して上で説明したような、環境１００の一部を照明するライト２０８を備えた制御可能なライト・システム１０２とすることができる。

【００３４】

一般に、ライト・システム１０２は、環境１００内にいて見ている人が、ライト・システム１０２によって投射された照明を直接見られるように、あるいは見ている人が、照明が表面から跳ね返った後や、またはレンズ、フィルタ、光学系、筐体、スクリーン、またはライト・システム１０２からの照明を反射、拡散、屈折、回折、または他の手段により照明に影響を及ぼすように設計されている類似の構成要素を通してなど、照明を間接的に見られるように取り付けることができる。

【００３５】

ライト・システム１０２は、組み合わせることで、照明またはイルミネーション・システムをなす。照明システムは、有線接続、ケーブル接続、赤外線（ＩＲ）接続、無線周波数（ＲＦ）接続、他の種類の接続、またはこれらの組み合わせなどを含むが、これらに限

定されない、当業者であれば知っているような方法で、制御システム、またはコンピュータなどの他のユーザ・インターフェース２０２と通信することができる。

【００３６】

後述のように、さまざまな制御システムを使用することにより照明制御信号を発生することができる。一実施形態では、ビデオ・ツー・ＤＭＸデバイスを介して照明システムに制御権を渡すことができ、これは、照明信号を発生する簡便な手段となる。このようなデバイスは、ビデオ入力ポートおよびパススルー・ビデオ出力ポートを備えることができる。このデバイスはさらに、照明信号ポートを備え、これにより、ＤＭＸまたはその他のプロトコル・データを室内に配備されているライトに通信する。デバイスは、あるアルゴリズムを、受信したビデオ信号（例えば、平均、所与のセクションまたは期間の平均、最大、最小）に適用し、アルゴリズムの出力に対応する照明信号を発生することができる。例えば、このデバイスでは、１秒間の信号の平均を取ることができ、得られる値は青色の光に等しい。次に、このデバイスは、青色の光信号を発生し、それを照明システムに通信することができる。一実施形態では、単純なシステムであれば、平均を取った同じ信号を室内の全ライトに通信するが、変形では信号の一部の平均を室内の一部に通信する。ビデオ信号を分割する方法は多数あり、照明システムのさまざまなセクションにアルゴリズムを適用し、同じビデオ信号に基づいて異なる入力を供給することが可能である。

【００３７】

また図１を参照すると、環境１００は１つまたは複数のライト・システム１０２により照明される表面１０７を含む。図示の実施形態では、表面１０７は、光が反射できる壁またはその他の表面を含む。他の実施形態では、表面は、場合によっては異なる周波数で、光を吸収し、そして再び送るように設計することができる。例えば、表面１０７は、燐光体でコーティングされたスクリーンとし、特定の色の照明をそのスクリーンに投射し、スクリーンではその照明の色を変換して、異なる色の照明を環境１００内の見ている人に送る。例えば、投射される照明は主に、青色、紫色、紫外線範囲の光であるが、送られる光線はよりたいてい白色である。複数の実施形態では、表面１０７はさらに、１つまたは複数の色、図、線、図案、像、絵、写真、テクスチャ、形状、または照明システムにより照明されるその他の視覚的要素あるいはグラフィック要素を含むことができる。この表面上にある要素は、テクスチャ、素材、コーティング、塗装、染料、顔料、被覆、織物、またはグラフィック効果または視覚的効果を出すその他の方法あるいはメカニズムによって作成することができる。複数の実施形態では、照明システムからの照明を変更することにより、視覚的効果が得られる。例えば、表面１０７に置かれている絵は、表面１０７に映し出される照明システムからの光の色に基づいて、次第に薄くなってゆくか、または消えるか、またはよりはっきりするか、または再表示することができる。したがって、表面１０７上に生成されることができる効果は、平面上に光を当てるだけでなく、表面上の視覚的要素またはグラフィック要素との光の相互作用を利用することでも行える。

【００３８】

いくつかの好ましい実施形態では、ライト・システム１０２はネットワーク化された照明システムであり、そこにおいては、照明制御信号をアドレス指定された情報のパケットに組み込む。次に、アドレス指定された情報は、照明ネットワーク内の照明システムに通信することができる。照明システムはそれぞれ、特定の照明システムにアドレス指定された制御信号に応答することができる。これは、幾つかの照明システムにまたがって照明効果を発生し調整するのに極めて有効な配置である。米国特許出願第０９／６１６２１４号「照明シークエンスをオーサリングするシステム及び方法（Ｓｙｓｔｅｍｓ ａｎｄ Ｍｅｔｈｏｄｓ ｆｏｒ Ａｕｔｈｏｒｉｎｇ Ｌｉｇｈｔｉｎｇ Ｓｅｑｕｅｎｃｅｓ）」の実施形態は、システム制御信号を発生するシステムおよび方法を説明しており、参照により本明細書に組み込まれている。

【００３９】

本発明の原理による照明システムまたは別のシステムをアドレス指定可能なコントローラに関連付けることができる。アドレス指定可能なコントローラは、そのアドレスが「聞

こえる」までネットワーク情報に「耳を傾ける」ように配置することができる。一度システム・アドレスが識別されると、システムは、そのアドレスに割り当てられたデータパケット内の情報を読み込んで、応答することができる。例えば、照明システムはアドレス指定可能なコントローラを備えることができる。さらにアドレス指定可能なコントローラは、変更可能アドレスを持つため、ユーザ側でシステムのアドレスを設定することができる。照明システムは、ネットワーク情報が通信されるネットワークに接続することができる。ネットワークを使用することにより、例えば、複数の照明システムなどの多くの制御されたシステムに情報を通信することができる。このような配置では、複数の照明システムのそれぞれが２以上の照明システムに関係する情報を受信することができる。情報は、第１のアドレス指定された照明システムのための情報の後に第２のアドレス指定された照明システムを宛先とする情報が続く、ビット・ストリームの形を取り得る。このような照明システムの一例が、米国特許第６０１６０３８号で取りあげられており、参照により本明細書に組み込まれている。

【００４０】

図１１の本発明の原理によるネットワーク化された照明システムの一実施形態では、ネットワーク送信機１１０２はネットワーク情報をライト・システム１０２に通信する。このような実施形態では、ライト・システム１０２は入力ポート１１０４および出力ポート１１０８を備えることができる。ネットワーク情報は、第１のライト・システム１０２に通信され、第１のライト・システム１０２は、そのシステムに対しアドレス指定されている情報を読み込み、その情報の残り部分を次のライト・システム１０２に受け渡すことができる。当業者であれば、本発明の原理によるシステムが包含するネットワークトポロジがほかにもあることを理解するであろう。

【００４１】

一実施形態では、ライト・システム１０２は現実世界の環境１００内に配置されている。現実世界の環境１００は、部屋とすることもできる。例えば、室内の壁、天井、床、またはその他のセクションまたは物、あるいは部屋の特定の表面１０７を照明するように照明システムを配置することも可能である。照明システムは、個別アドレスを有する幾つかのアドレス指定可能なライト・システム１０２を備えることができる。照明は、部屋の中の見る人が直接または間接的に見ることができるように投射することができる。つまり、ライト・システム１０２の光２０８は、反射させずに見ている人に投射されるように光らせたり、反射、屈折、吸収、および再放射するか、または他の手段により間接的に見ている人に見せることができる。

【００４２】

本発明の一実施形態では、図３のブロック図に示されているような制御信号を発生する方法について説明する。この方法では、イメージまたはイメージの表現、つまりグラフィック表現３０２を供給または発生する必要がある。グラフィック表現は、図面、写真、発生されたイメージなどの静的なイメージ、または静的であるイメージ、あるいは静的であるように見えるイメージとすることができる。静的なイメージは、スクリーン上で継続的に更新（リフレッシュ）されているとしてもコンピュータ・スクリーンまたはその他のスクリーン上に表示されるイメージとしてよい。静的なイメージは、さらにイメージのハード・コピーでもよい。

【００４３】

グラフィック表現３０２を実現するにはさらに、イメージを発生またはイメージの表現を発生する必要がある。例えば、プロセッサを使って、グラフィック表現３０２を発生するソフトウェアを実行することができる。ここでもまた、発生されるイメージは、静的であっても、静的であるように見えてもよく、あるいは動的なイメージであってもよい。動的イメージを発生するために使用するソフトウェアの一例として、Ｍａｃｒｏｍｅｄｉａ，Ｉｎｃｏｒｐｏｒａｔｅｄが販売しているＦｌａｓｈ５・コンピュータソフトウェアがある。Ｆｌａｓｈ５は、グラフィックス、イメージ、およびアニメーションを発生するために広く使用されているコンピュータ・プログラムである。イメージを発生するために使

用される有用な製品としては他に、例えば、Ａｄｏｂｅ　Ｉｌｌｕｓｔｒａｔｏｒ、Ａｄｏｂｅ　Ｐｈｏｔｏｓｈｏｐ、およびＡｄｏｂｅ　ＬｉｖｅＭｏｔｉｏｎなどがある。静的イメージと動的イメージの両方の発生に使用できるプログラムはほかにも多数ある。例えば、Ｍｉｃｒｏｓｏｆｔ　Ｃｏｒｐｏｒａｔｉｏｎで提供しているコンピュータ・プログラムＰａｉｎｔがある。このソフトウェアは、スクリーン上にビットマップ形式のイメージを発生する場合に使用する。ビットマップ、ベクトル座標、またはその他の手法でイメージを発生するために他のソフトウェア・プログラムを使用することもできる。また、３次元またはそれ以上の次元でグラフィックスを描画するプログラムも多数ある。例えば、Ｍｉｃｒｏｓｏｆｔ　ＣｏｒｐｏｒａｔｉｏｎのＤｉｒｅｃｔ　Ｘライブラリを使用すると、イメージを３次元空間内に発生することができる。前記のソフトウェア・プログラムまたは類似のプログラムからの出力をグラフィック表現３０２として使用することができる。

【００４４】

複数の実施形態では、グラフィック表現３０２をプロセッサで実行されるソフトウェアを使用して発生し、グラフィック表現３０２を全くスクリーン上に表示しないようにすることができる。一実施形態では、アルゴリズムにより、例えば、室内での爆発などのイメージまたはイメージの表現を発生することができる。この爆発機能により、イメージを発生し、そのイメージを使用して、本明細書で説明しているような制御信号を発生することができるが、イメージを実際にスクリーン上に表示することも表示しないこともできる。例えば、イメージをスクリーン上に表示することなく、照明ネットワークを通じて表示することができる。

【００４５】

一実施形態では、イメージを発生または表現するには、プロセッサ上で実行されるプログラムを利用する。一実施形態では、イメージまたはイメージの表現を発生する目的は、空間内で定義された情報を提供することにある。例えば、イメージの発生により、室内への照明効果の伝わり方を定めることができる。照明効果は例えば爆発を表す。この表現により、部屋の隅での明白色光のオンが開始し、光はこの部屋の隅からある速度（速さと方向を持つ）で進行し、光の色は効果の伝搬が続くにつれ変化する。いくつかの照明効果の速度を表すベクトル１０４を示す環境１００が、図１に示されている。一実施形態では、イメージ発生器により関数またはアルゴリズムが発生される。関数またはアルゴリズムは、爆発、照明の当たり、ヘッドライト、室内の列車通過、室内を通る弾丸発射、室内の光の移動、室内の日の出などの事象、またはその他の事象を表すことができる。関数またはアルゴリズムは、室内での光の回転、室内の光のボールの跳ね返り、室内の音の跳ね返りなどのイメージ、またはその他のイメージを表すことができる。関数またはアルゴリズムは、さらに、ランダムに発生された効果またはその他の効果を表すこともできる。

【００４６】

再び図３を参照すると、ライト・システム構成機構３０４は、本明細書で説明している方法およびシステムのための他のステップを実行することができる。ライト・システム構成機構は、図１と関連して説明しているものなどの照明システムのためのシステム構成ファイル、構成データ、またはその他の構成情報を発生することができる。

【００４７】

ライト・システム構成機構は、ライト・システム１０２、本明細書で説明しているような、音響システムなどのシステムまたは他のシステムを表し、又はそれらを、環境１００内の位置または複数の位置と相関関係を持つことができる。例えば、ＬＥＤライト・システム１０２は、室内の位置と相関関係を持たせることができる。一実施形態では、照明されている表面１０７の位置も、構成ファイルの中に含むため決定されることができる。照明されている表面の位置をさらに、ライト・システム１０２と関連付けることができる。複数の実施形態では、照明されている表面１０７は所望のパラメータとすることができるが、表面を照明する光を発生するライト・システム１０２も重要である。ライト・システム１０２により表面に照明を当てるスケジュールを立てている場合、照明制御信号をライ

ト・システム１０２に通信する。例えば、発生されたイメージについて部屋の特定のセクションの色調、彩度、または明度を変える必要があるときに制御信号を照明システムに通信する。このような状況で、制御信号を使用することにより、照明システムを制御し、光が当てられる表面１０７が適切な時期に照明されるようにすることができる。照明される表面１０７は、壁に位置されることができるが、光を表面１０７上に投射するように設計されているライト・システム１０２は天井に配置することができる。構成情報は、表面１０７を照明する場合にライト・システム１０２を起動して、アクティブ化または変更するように構成することができる。

【００４８】

さらに図３を参照すると、グラフィック表現３０２、およびライト・システム構成機構３０４からの構成情報を変換モジュール３０８に送り、その変換モジュール３０８は、構成機能からの位置情報をグラフィック表現の情報に関連付け、その情報をライト・システム１０２用の制御信号などの制御信号に変換することができる。次に、変換モジュールは、制御信号をライト・システム１０２などに通信することができる。いくつかの実施形態では、変換モジュールは、グラフィック表現内の位置を、環境用の構成ファイルに格納されているような環境内のライト・システム１０２の位置にマッピングする（後述）。このマッピングは、グラフィック表現内のピクセルまたはピクセルのグループを環境１００内のライト・システム１０２またはライト・システム１０２のグループに対応付ける１対１のマッピングとすることができる。これは、グラフィック表現内のピクセルを、ライト・システム１０２により照明される環境内の表面１０７、多角形、または対象物に対応付けるマッピングとすることもできる。これは、ベクトル座標情報、波動関数、またはアルゴリズムをライト・システム１０２の位置に対応付けるマッピングとすることもできる。多くの異なるマッピング対応関係を思いつくことができるため、それらも本明細書に含まれる。

【００４９】

図４を参照すると、制御信号を発生する方法およびシステムのブロック図の他の実施形態が示されている。ライト管理機構４０２は、ライト・システム１０２を環境内の位置、ライト・システムによって照明される表面などにマッピングするマップ・ファイル４０４を発生するために使用される。アニメーション機構４０８では、アニメーション効果のための一連のグラフィックス・ファイルを発生する。変換モジュール４１２は、ライト・システム１０２のためのマップ・ファイル４０４内の情報をグラフィックス・ファイル内のグラフィック情報に関連付ける。例えば、グラフィックス・ファイル内の色情報を使用することにより、色制御信号に変換して、ライト・システムが類似の色を発生できるようにする。グラフィックス・ファイルのためのピクセル情報は、注目しているピクセルに対応するライト・システムのためのアドレス情報に変換することができる。複数の実施形態では、変換モジュール４１２は、照明システムのための構成ファイルの内容および注目しているアニメーション機構に好適な変換アルゴリズムに基づいて、特定のグラフィックス・ファイル情報を特定の照明制御信号に変換するためのルックアップ・テーブルを備える。変換された情報は、再生ツール４１４に送られ、次に、アニメーションが再生され、制御信号４１８が環境内のライト・システム１０２に送られる。

【００５０】

図５を参照すると、構成ファイル５００の一実施形態が示されており、そこでは、構成情報のいくつかの要素を、ライト・システム１０２または他のシステム用に格納できることが説明されている。したがって、構成ファイル５００は、それぞれのライト・システム１０２のための識別子５０２だけでなく、環境１００に対する所望の座標またはマッピング・システム内のその照明システムの位置５０８も格納することができる（（ｘ，ｙ，ｚ）座標、極座標、（ｘ，ｙ）座標など）。位置５０８およびその他の情報は、時間に依存し、そのため構成ファイル５００は時間の要素５０４を含むことができる。構成ファイル５００はさらに、ライト・システム１０２によって照明される位置５１０に関する情報を格納することもできる。その情報は、一組の座標とすることもできるが、又はそれは、環

境内の識別された表面、多角形、対象物、またはその他の品物であってもよい。構成ファイル５００はさらに、色範囲５１２内の利用可能な色、光度範囲５１４内の使用可能な光度など、ライト・システム１０２を使用する上での使用可能な自由度に関する情報を格納することもできる。構成ファイル５００はさらに、本明細書で開示されている制御システムにより制御される環境内の他のシステムに関する情報、環境内の表面１０７の特性に関する情報なども含むことができる。したがって、構成ファイル５００は、一組のライト・システム１０２を、それらが環境１００内で発生することが可能な条件にマッピングすることができる。

【００５１】

一実施形態では、構成ファイル５００などの構成情報は、プロセッサ上で実行されるプログラムを使用して発生することができる。図６を参照すると、プログラムを、グラフィカル・ユーザ・インターフェース６１２を用いてコンピュータ６００上で実行し、環境６０２の表現を表示し、ライト・システム１０２、照明されている表面１０７、または他の要素をグラフィック形式で表示することができる。例えば、インターフェースには部屋の表現６０２を含めることができる。次に、ライト、照明されている表面、または他のシステムの表現をインターフェース６１２内に表示し、配置をシステムに割り当てることができる。一実施形態では、位置座標または位置マップは、ライト・システムなどのシステムを表す。また、例えば、照明されている表面の表現に対する位置マップを発生することもできる。図６は、ライト・システム１０２を備える部屋を示している。

【００５２】

表現６０２は、効果の発生を簡略化するためにも使用できる。例えば、一組の格納されている効果をスクリーン６１２上のアイコン６１０により表すことができる。爆発アイコンをカーソルまたはマウスで選択すると、ユーザは座標系内の爆発の開始点と終了点をクリックするよう求められる。表現内にベクトルを配置することにより、ユーザは、部屋６０２の上側隅で爆発を引き起こし、光波および／または音を環境内に伝搬させることができる。構成ファイル５００の中に識別されているように、ライト・システム１０２の全てを所定の位置に配置すると、爆発の表現をライト・システム、および／または音響システムなどの他のシステムにより部屋内で再生することができる。

【００５３】

使用時に、本明細書で使用しているような制御システムを使用することにより、コンピュータ６００のユーザに与えられる情報に応答して、またはそれに合わせて、情報をライト・システム１０２からユーザまたはプログラマに与えることができる。これを与える方法の一例は、ユーザがコンピュータ６００上でコンピュータ・アニメーションを発生することと関連する。ライト・システム１０２を使用することにより、コンピュータ６００上の表示６１２に応答して、１つまたは複数の光効果を生成することができる。照明効果、またはイルミネーション効果により、色が変化する効果、ストロボスコープ効果、フラッシュ効果、調和した照明効果、ビデオまたはオーディオなどの他の媒体と調和する照明効果、色が一定期間の間に色調、彩度、または光度を変えるカラー・ウオッシュ（ｃｏｌｏｒ　ｗａｓｈ）、周囲の色を生成する効果、退色効果、カラー・チェーシング・レインボー（ｃｏｌｏｒ　ｃｈａｓｉｎｇ　ｒａｉｎｂｏｗ）、室内のフレア・ストリーキング（ｆｌａｒｅ　ｓｔｒｅａｋｉｎｇ）、日の出、爆発の噴流、その他の移動効果などの移動をシミュレートする効果、およびその他多数の効果を含む膨大な数の効果を生み出すことができる。出せる効果は、無限と言っていいくらいである。光および色がユーザを連続的に取り囲み、空間内の照明または色を制御または変化させることにより、感情を変化させ、雰囲気を作り出し、素材や対象物の増強を行い、あるいは他の楽しいまたは有用な効果を生み出す。コンピュータ６００のユーザは、表示６１２上で効果を修正しながらその効果を観察できるため、ユーザが楽に効果を修正できるフィードバック・ループが可能である。

【００５４】

図７は、所与のライト・システム１０２からの光が表面上にどのように表示されるかを

示している。ライト・システム１０２、音響システム、またはその他のシステムは表面への投影を行うことができる。ライト・システム１０２の場合、これは、ライト・システム１０２によって照明される範囲７０２とすることができる。ライト・システム１０２、またはその他のシステムもまた移動できるため、範囲１０７も移動できる。音響システムの場合、これは、ユーザ側で音を放射したい範囲であるとしてよい。

【００５５】

一実施形態では、イメージまたは表現を形成するために発生される情報をライト・システム１０２または複数のライト・システム１０２に通信することができる。情報は、構成ファイルの中に発生されたように照明システムに送信される。例えば、イメージは室内の右上隅から始まる爆発を表し、その爆発は室内を伝搬する。イメージが計算された空間内を伝搬してゆくにつれ、制御信号を対応する空間内の照明システムに通信することができる。通信信号により、イメージが照明システムの投射先である照明されている空間を通過しているときに照明システムは所与の色調、彩度、および光度の光を発生することができる。本発明の一実施形態では、イメージを照明システムを通して投影する。イメージは、さらに、コンピュータ・スクリーンまたはその他のスクリーンまたは投射デバイスを介して投射することもできる。一実施形態では、イメージを照明システム上で再生する前または再生しているときにスクリーンを使用してイメージを視覚化することができる。一実施形態では、音響または他の効果を照明効果と相関関係付けし得る。例えば、空間内を伝搬する光波のピーク強度を音波のすぐ前に来るように設定することができる。その結果、光波が室内を通過するのに続いて音波が通過することになる。光波は照明システムで再生され、音波は音響システムで再生される。この調和により、室内を通過してゆくようにみえる効果を生み出したり、他の種々の効果を生み出すことができる。

【００５６】

図６では、効果は、コンピュータ６００の表示スクリーン６１２上に３Ｄ表示される仮想環境内を伝搬することができる。複数の実施形態では、効果を時間の経過とともに空間内を移動するベクトルまたは平面としてモデル化することができる。そこで、現実世界の環境における効果の平面上に配置されているすべてのライト・システム１０２を制御することにより、効果平面がライト・システム平面内を伝搬してゆくときにある種の照明を発生するようにできる。これは、表示スクリーンの仮想環境内でモデル化され、そのため開発者は時間の経過とともに変化する一連の位置を辿って平面をドラッグすることができる。例えば、効果平面６１８は、ベクトル６０８が仮想環境内を通過するとともに移動する。効果平面６１８が多角形６１４に到達すると、その多角形はカラー・パレット６０４から選択された色で強調表示される。そこで、その多角形に対応する現実世界の対象物上に配置されているライト・システム１０２は、現実世界の環境と同じ色で照明することができる。もちろん、多角形は、対象物、平面、表面、壁などに配置される任意の構成のライト・システムとすることができ、作成できる３Ｄ効果の範囲は無制限といえる。

【００５７】

一実施形態では、中央コントローラからイメージ情報を通信することができる。情報は、照明システムがその情報に応答する前に変更することができる。例えば、イメージ情報を位置マップ内のある位置を宛先として送ることができる。情報を照明システムに送信する前に、位置マップで宛先が指定されているすべての情報を収集することができる。この作業は、イメージが更新される毎に、またはイメージのこのセクションが更新される毎に、または他のときに実行できる。一実施形態では、収集された情報に対しアルゴリズムを実行することができる。このアルゴリズムで行うのは、情報の平均、最大値情報の計算および選択、最小値情報の計算および選択、情報の第１四分位値の計算および選択、情報の第３四分位値の計算および選択、最も使用される情報の計算および選択、情報の積分の計算および選択、または情報に対する別の計算の実行である。このステップを実行して、受け取った情報に応答して照明システムの効果を均す。例えば、一回の更新サイクルでの情報によりマップ内の情報が数回変更され、そして投射された光が所与の更新サイクルで１つの値を取るときにその効果を最良の形で表示することができる。

【００５８】
一実施形態では、照明システムに通信される情報は、照明システムが情報に応答する前に変更することができる。情報の形式は、例えば通信前に変えることができる。情報は、ＵＳＢポートまたはその他の通信ポートを通じてコンピュータから通信することができ、そして情報の形式は、照明システムへの情報の通信時にＤＭＸなどの照明プロトコルに変更できる。一実施形態では、情報または制御信号は、コンピュータ、携帯型コンピュータ、ノートブック・コンピュータ、パーソナル・ディジタル・アシスタント（ＰＤＡ）、またはその他のシステムの通信ポートを介して照明システムまたはその他のシステムに通信することができる。また、情報または制御信号は、電子的形態またはその他の形態のメモリに格納しておき、後で取り出すことができる。Ｃｏｌｏｒ Ｋｉｎｅｔｉｃｓ Ｉｎｃｏｒｐｏｒａｔｅｄが製造販売しているｉＰｌａｙｅｒおよびＳｍａｒｔＪａｃｋシステムを使用すると、照明制御信号を通信および／または格納することができる。
【００５９】
一実施形態では、いくつかのシステムを位置マップに関連付けることができ、これらのいくつかのシステムは位置マップを共有するか、またはシステムを独立の位置範囲に配置することができる。例えば、第１の照明システムからの照明されている表面の位置は第２の照明システムからの照明されている表面と交差し得る。この２つのシステムは、それでも、照明システムのいずれかに通信される情報に応答することができる。一実施形態では、２つの照明システムの相互作用も制御することができる。アルゴリズム、関数、またはその他の手法を使用して、インタラクティブな空間内の１つまたは複数の照明システムの照明効果を変更することができる。例えば、インタラクティブな空間が照明システムからの非インタラクティブな空間の半分を超える場合、照明システムの色調、彩度、または明度を変更して、インタラクティブな範囲を補償することができる。これを用いて、インタラクティブな範囲または例えば隣接する範囲の全体的な外観を調整することができる。
【００６０】
本発明の原理による方法および／またはシステムを使用して発生される制御信号を使用することにより、膨大な種類の効果を生み出すことができる。火または爆発効果を壁または部屋にわたって動かしたい場合がある。これは、部屋の一端から、白色閃光として始まって、たちまち出ていってしまい、その後に高輝度の黄色の光波が続き、その光度は室内での移動に従って変化する。本発明の原理により制御信号を発生する場合に、照明デザイナは、室内のライト、およびそれぞれの照明システムの照明効果のタイミングおよび発生を気にかける必要はない。むしろ、照明デザイナは、室内のライトの相対的位置または実際の位置だけを気にすればよい。照明デザイナは、室内の照明のレイアウトを決定し、その後、室内のライトを上述のようにピクセル情報などのグラフィック情報に関連付けることができる。照明デザイナは、例えば、Ｆｌａｓｈ５を使用して、コンピュータ上で火または爆発効果をプログラムし、そして環境内のライト・システム１０２にその情報を通信することができる。環境内のライトの位置だけでなく、照明されることになる表面１０７または範囲７０２も考慮することができる。
【００６１】
一実施形態では、照明効果はさらに、音に結合し、音を照明効果に付け加え、照明効果を高めることができる。例えば、「ウ～、ウ～」というサイレンのような効果が部屋全体に結合され、音に合わせて赤色で明滅する「赤警報（非常警報）」シーケンスがある。１つの刺激で他の刺激が強められる。低周波の音およびちらちらする光を使用した地震の音と動きは、これらの効果を調和するもう１つの例である。光および音の動きを使用することにより方向を示すことができる。
【００６２】
一実施形態では、ライトは二次元つまり平面内に表現される。これにより、ライトを平面内に表現し、ライトをさまざまなピクセルに関連付けることができる。標準のコンピュータ・グラフィックス手法を効果に使用することができる。照明効果を生成するために、アニメーション・トゥイーニング（ａｎｉｍａｔｉｏｎ ｔｗｅｅｎｉｎｇ）、さらには

標準的なツールさえも使用することができる。Ｍａｃｒｏｍｅｄｉａ　Ｆｌａｓｈは、ウェブ上でアニメーションを作成するために比較的低解像度のグラフィックスを使用する。Ｆｌａｓｈは、単純なベクトル・グラフィックスを使用してアニメーションを簡単に作成する。ベクトル表現は、アニメーションをネット越しに送信するためワールド・ワイド・ウェブ（Ｗｏｒｌｄ　Ｗｉｄｅ　Ｗｅｂ）などでアプリケーションのストリーミングを行う場合に効率がよい。同じ技術を使用して、ピクセル情報またはベクトル情報を、環境１００の座標系内のライト・システム１０２の位置に対応するベクトルまたはピクセルにマッピングすることにより照明コマンドを導き出すために使用できるアニメーションを生成することができる。

【００６３】

例えば、コンピュータ６００のアニメーション・ウィンドウは、部屋またはライトの他の環境を表すことができる。そのウィンドウ内のピクセルを部屋の中のライトに対応付けるか、または低解像度の平均化されたイメージをより高い解像度のイメージから生成することができる。このようにして、対応するピクセルまたはピクセルの近傍がオンしたときに部屋の中のライトを活動状態にすることができる。ＬＥＤベースの照明技術では、要求があった場合にディジタル制御情報を使用してどのような色も創り出せるため（米国特許第６０１６０３８号、第６１５０７７４号、および第６１６６４９６号を参照）、ライトは、元のイメージの色を忠実に再現できる。

【００６４】

本発明の原理によるシステムおよび方法を使用して発生することが可能ないくつかの効果の例は、爆発、色、水中効果、乱流、色の変化、火、ミサイル、追跡、部屋の回転、形状の運動、ティンカーベルのような形、室内を移動する光、さらに他の多くの効果を含むが、これらに限定されない。効果はいずれも、周波数、波長、波の幅、ピーク・ツー・ピーク測定値、速度、慣性、摩擦、速さ、幅、スピン、ベクトルなどのパラメータで指定することができる。これらはいずれも音響などの他の効果に結合することができる。

【００６５】

コンピュータ・グラフィックスでは、アンチエイリアシングが、エッジを描画し且つ解像度が制限されているイメージ内の階段効果を除去するための１つの手法となっている。この効果は、テレビジョンに幅の狭い縞模様が現れる場合に見られる効果である。エッジは、線が水平に近づくとともに蟻が這っているように見える。同様にして、効果が動いているときに滑らかな遷移になるような方法で照明を制御することができる。波の幅、振幅、位相、または周波数などの効果パラメータを修正して、効果を改善することができる。

【００６６】

例えば、図８を参照すると、概略図８００に時間の経過によって単一のライト８０４を表す円が描かれている。このライトを「トラバースする」効果については、単にステップ関数があり、これにより波がライトを通過するときに光がパルスを発生するようにできる。しかし、幅という概念がないと、効果の違いを見分けることはできない。効果には幅があるのが好ましい。しかし、光の効果が単に、一定期間オンしているステップ関数であった場合、荒っぽい遷移のように見え、場合によってはそれが望ましいこともあるが、時間の経過とともに移動する効果であれば（つまり、何らかの速度が関連付けられている）、通常そのような遷移であってはならない。

【００６７】

図８に示されている波８０２は、変化に対応する形を持つ。本質的に、空間内を伝搬するときに生じる波８０２の視覚的な畳み込みである。したがって、爆発によるものなどの波が空間内のいくつかの点を過ぎるときに、それらの点の光度が０から上昇し、さらには、色調または彩度に関連する変化さえあり、効果の動きに対してより現実味を帯びた効果が得られる。ある点で、ライトの個数および密度が増えるにつれ、部屋はスクリーンの延長となり、大きくまばらなピクセルが得られる。ライト・システム１０２が比較的少数でも、効果は最終的に、大きなスクリーン表示に類似の表示に使用することができる。

【００６８】

効果は、関連する運動および方向、つまり速度を持つことができる。他の物理パラメータであっても、摩擦、慣性、およびモーメントなどの物理パラメータを与えるため記述することができる。それもさることながら、効果は特定の軌道を持つことができる。一実施形態では、各ライトは、ライトの属性を与える表現を持つ。これは、例えば、２Ｄ位置の形態を取ることができる。ライト・システム１０２は、さまざまなすべての自由度を割り当てたり（例えば、ｘｙｚ－ｒｐｙ）、任意の組み合わせを使用することができる。

【００６９】

ここに取りあげている手法は、照明に限られない。制御信号は、特殊効果装置、例えば、花火、匂い発生装置、霧発生装置、泡発生装置、移動機構、音響装置、空間内を動く音響効果、またはその他のシステムなど、それらの位置に基づく他の装置を介して伝搬できる。

【００７０】

本発明の一実施形態は、環境内のライト・システム１０２の位置を自動的に捕捉する方法である。像形成装置を、ライトの位置を捕捉する一手段として使用することができる。コンピューティング・デバイスに接続されているカメラにより、分析対象のイメージを捕捉して、ライトの位置を計算することができる。図９は、この方法を実行するために使用できる一連のステップを説明する流れ図９００である。まず、ステップ９０２で、周囲光を減光することによりマッピングの対象の環境を暗くする。次に、ステップ９０４で、制御信号を各ライト・システム１０２に送信し、ライト・システム１０２にオン及びオフを順に行うよう指令する。それと同時に、ステップ９０６で、カメラは、それぞれの「オン」のときのイメージを捕捉する。次に、ステップ９０８で、イメージを分析して、「オン」のライト・システム１０２の位置を特定する。ステップ９１０で、重心を抽出することができる。特定のライト・システム１０２がオンしている場合には他のライトが存在していないため、イメージにフィルタ処理を行い取り除くべき他のアーチファクトについての問題はほとんどない。その次に、ステップ９１２で、ライト・システム１０２の重心位置を格納し、そしてシステムはライト・システム１０２と重心位置のテーブルを発生する。このデータを使用して、図５に関して説明したような構成ファイルに値を設定する（ｐｏｐｕｌａｔｅ）ことができる。つまり、各ライト・システム１０２は順に、活動状態にされ、中心測定値が決定されるということである。これは、ライト・システム１０２のすべてについて行う。したがって、イメージにより、（ｘ，ｙ）座標などの、平面内の照明システムの位置が得られる。

【００７１】

３Ｄ位置が必要な場合、第２のイメージを捕捉して、別の座標次元でライトの位置の三角測量を行うことができる。これはステレオの問題である。同じようにして、人間が目で見て得られるイメージ間の対応関係と食い違いとを利用して深さを決定し、第２のイメージの集まりを取り出して対応関係を定めることができる。カメラを第１のカメラに関して知られている位置に複製するか、または第１のカメラを固定の距離および方向で移動する。この移動または位置の違いにより、２つのイメージのベースラインが確立し、ライト・システム１０２のための第３の座標（例えば、（ｘ，ｙ，ｚ））を導くことができる。

【００７２】

本発明の他の実施形態は、制御信号を発生するステップをまとめた流れ図１０００が示されている図１０で説明されている。まず、ステップ１００２で、ユーザは、図６に示されている表示６１２などのグラフィカル・ユーザ・インターフェースにアクセスする。次に、ステップ１００３で、グラフィックス・プログラムまたは類似の機能を使用するなどして表示装置上にイメージを発生することができる。イメージは、部屋、壁、建物、表面、対象物など、ライト・システム１０２が配置されている環境の表現とすることができる。図１０と関連して、環境内のライト・システム１０２の構成が知られており、テーブルまたは構成ファイル５００に格納されていると仮定する。次に、ステップ１００４で、ユーザは、効果のメニューなどを使用して、効果を選択することができる。一実施形態では、効果はカラー・パレットから選択した色であってよい。色は、白色の色温度でよい。効

果は、本明細書で説明しているような別の効果でもよい。一実施形態では、イメージ１００３を発生するには、プロセッサ上で実行されるプログラムを利用する。イメージはさらに、コンピュータ・スクリーン上に表示することもできる。ステップ１００４でカラー・パレットから色を選択した後、ユーザはステップ１００８でイメージの一部を選択することができる。この作業は、グラフィカル・ユーザ・インターフェースのスクリーン上のカーソルを使用して行うことができ、そこではイメージの所望の部分の上にカーソルを置き、その部分をマウスで選択する。イメージの一部を選択した後、ステップ１０１０でその部分から得られる情報を照明制御信号に変換することができる。この作業では、ビット・ストリームの形式を変更するか、またはその情報を他の情報に変換する必要がある。イメージを作成するのに使用する情報は、赤色、緑色、および青色などのいくつかの色に分割する（ｓｅｇｍｅｎｔ）ことができる。情報はさらに、例えば、セグメント分割された赤色、緑色、および青色信号で照明システムに通信することもできる。信号は、さらに、ステップ１０１２で複合信号として照明システムに通信することもできる。この手法は、照明システムの色を変更する場合に有用である。例えば、カラー・パレットをグラフィカル・ユーザ・インターフェース内に表示し、そしてそのパレットを使って何百万種類もの色を表現することができる。ユーザは、室内または他の範囲の照明を藍色に変えたい場合がある。そうするには、マウスを使ってスクリーンから色を選択すると、室内の照明を選択したスクリーンの部分の色に合うように変更できる。一般に、コンピュータ・スクリーン上の情報は、赤色、緑色、および青色の小さなピクセルで表される。米国特許第６０１６０３８号、第６１５０７７４号、および第６１６６４９６号に記載されているＬＥＤシステムには、赤色、緑色、および青色の照明構成要素も含まれる。スクリーン上の情報から制御信号への変換プロセスは、照明システムがコマンドを理解できるような形式変更である。しかし、一実施形態では、別々の照明構成要素のレベルに関する情報は、ピクセル情報を発生するために使用される情報と同じでよい。これにより、照明システムでピクセル情報を正確に複製することができる。

【００７３】

環境内のライト・システムの位置を決定する手法、環境内の効果をモデル化する手法（時間に関する効果および幾何学的な効果を含む）、およびライト・システムの環境を仮想環境にマッピングする手法をはじめとする、本明細書で説明している手法を使用することにより、無数の環境において無数の効果をモデル化することが可能である。効果は、正方形または矩形の表示上で描画できる効果に限られるわけではない。そうではなく、ライト・システムを広い範囲にわたる線、ストリング、曲線、多角形、円錐、円柱、立方体、球体、半球、非線形構成、雲形、および任意の形状および構成で配置し、選択した座標次元でそれらの位置を捕捉する仮想環境でモデル化することができる。したがって、ライト・システムは、部屋、建物、家、壁、対象物、製品、小売店、車両、船舶、飛行機、プール、温泉、病院、手術室、またはその他の場所など、環境の内側または外側に配置することができる。

【００７４】

複数の実施形態では、ライト・システムにコンピュータ・アプリケーションのためのコードを関連付け、それによりライト・システムを制御するようにコンピュータ・アプリケーション・コードを修正または生成することができる。例えば、オブジェクト指向プログラミング手法を使用することにより、コンピュータ・コード内のオブジェクトに属性を付加し、そしてその属性を使用して照明システムの挙動を制御することができる。オブジェクト指向手法は当該技術分野では知られており、Ｔｉｍｏｔｈｙ Ｂｕｄｄ著「Ｉｎｔｒｏｄｕｃｔｉｏｎ ｔｏ Ｏｂｊｅｃｔ－Ｏｒｉｅｎｔｅｄ Ｐｒｏｇｒａｍｍｉｎｇ」などのテキストで説明されているため、開示全体を参照により本明細書に取り込んでいる。他のプログラミング手法を使用することにより、コンピュータ・アプリケーションに合わせて照明を行うよう照明システムに指令を出すこともでき、オブジェクト指向プログラミングは本明細書で説明している方法およびシステムを利用しやすくする当業者であれば理解できるさまざまなプログラミング手法のうちの１つであることを理解すべきである。

【００７５】
一実施形態では、開発者はコンピュータ・アプリケーションのオブジェクトにライト・システムの入力をアタッチすることができる。例えば、開発者は、アプリケーション・オブジェクトのコード構成、つまりオブジェクトに付加されるライト・システム１０２の抽象化を行うことができる。オブジェクトは、位置、速度、色、光度、またはその他の値などのさまざまな属性からなる。開発者は、コンピュータ・アプリケーションのコードの中のオブジェクトにインスタンスとして光を加えることができる。例えば、オブジェクトは、方向と速度などの属性を持つ、オブジェクト指向コンピュータ・アニメーション・プログラムまたはソリッド・モデリング・プログラム内のベクトルとすることが可能である。ライト・システム１０２は、コンピュータ・アプリケーションのオブジェクトのインスタンスとして追加することができ、そしてライト・システムは光度、色、およびさまざまな効果などの属性を備えることができる。したがって、コンピュータ・アプリケーション内にベクトルのオブジェクトを呼び出す事象が発生した場合、プログラムで実行されるスレッドにより、ライト・システムのプロセッサへの入力として使用されるコードを引き出すことができる。光は、ジオメトリ、配置、空間内位置を正確に表すか、または属性または特性（ｔｒａｉｔ）の値を表すか、または他の要素またはオブジェクトに関する指示を与えることができる。

【００７６】
図１２の流れ図１２００は、調和した照明を可能にする方法のステップを示している。ステップ１２０２で、プログラマは、例えば、オブジェクト指向プログラミング手法を使用して、コンピュータ・アプリケーションのオブジェクトのコーディングを行う。ステップ１２０４で、プログラミングによりアプリケーション内のオブジェクトのそれぞれについてインスタンスを作成する。ステップ１２０８で、プログラマは、光をインスタンスとしてアプリケーションの１つまたは複数のオブジェクトに加える。ステップ１２１０で、プログラマは、アプリケーション・コードで実行されるスレッドを与える。ステップ１２１２で、プログラマは、インスタンスとしてライトを有するオブジェクトから照明システムの入力コードを引き出すためのスレッドを用意する。ステップ１２１４で、ステップ１２１２でスレッドから引き出された入力信号をライト・システムに供給し、それにより照明システムがコンピュータ・アプリケーションから引き出されたコードに応答するようにする。

【００７７】
コンピュータ・アプリケーションのコードからライト・システム１０２にこのようなオブジェクト指向光入力を使用することにより、現実世界の環境においてさまざまな照明効果をコンピュータ・アプリケーションの仮想世界オブジェクトに関連付けることができる。例えば、多角形の爆発などの効果のアニメーションでは、音、閃光、運動、振動、およびその他の時間的効果などの光効果を多角形の爆発とともに付加することができる。さらに、ライト・システム１０２は、音生成装置、運動生成装置、霧発生機構、降雨機構、またはそのオブジェクトに関係する指示も生成することができるその他の装置を含む他の効果発生装置を備えることができる。

【００７８】
図１３を参照すると、流れ図１３００に、コンピュータ・スクリーンの仮想環境上の表現と現実の環境内のライト・システム１０２またはライト・システム１０２の集まりとの間で調和した照明を行うためのステップが示されている。複数の実施形態では、ライト・システム１０２を制御するためのプログラム・コードは、制御信号を供給するマシン上で実行される別々のスレッドを有する。ステップ１３０２で、プログラムはスレッドを開始する。ステップ１３０４で、スレッドは、仮想光、つまり仮想環境内の光を表すプログラム・コード内のオブジェクトのリストを通してできる限り頻繁に実行される。ステップ１３０８で、スレッドは、三次元数学演算を実行して、環境内の現実世界のどのライト・システム１０２がコンピュータ表現の仮想環境内のオブジェクトの座標系の基準点として投射される現実世界（例えば、選択した表面１０７）の基準点に近いかを決定する。したが

って、（０，０，０）位置は、現実の環境内の位置、およびコンピュータ・アプリケーションの表示装置のスクリーン上の一点とすることができる（例えば、表示の中心）。ステップ１３１０で、コードにより、仮想環境を、ライト・システム１０２を含む現実世界の環境にマッピングし、それによりコンピュータ・スクリーンの外部で発生する事象が基準点に関して、仮想オブジェクトおよび事象がコンピュータ・スクリーン上の基準点に対するのと類似したものとなるようにする。

【００７９】

ステップ１３１２で、この方法のホスト側でマッピング用のインターフェースを用意することができる。マッピング関数は、ドラッグ・アンド・ドロップ・インターフェースなどの単純なユーザ・インターフェースを使用して現実世界のライトをマッピングする、以下で説明しまた付録Ａでも説明するＤｉｒｅｃｔｌｉｇｈｔ　ＡＰＩで記述されている関数、例えば「ｐｒｏｊｅｃｔ－ａｌｌ－ｌｉｇｈｔｓ」で実行することができる。ライトの配置は、ライトを向ける表面ほど重要ではない。照明を反射する、または光を環境に送り返すのがこの表面であるため、マッピング・プログラムにとって最も重要なのはこの表面なのである。マッピング・プログラムは、照明システムの位置ではなくこれらの表面をマッピングするか、または照明システムの位置と表面上の光の位置の両方をマッピングすることもできる。

【００８０】

調和した照明を行うためのコードを供給するためのシステムは、プロセッサ、オペレーティング・システム、および実行のためファイルを格納するためのデータベースなどのメモリを備える、プログラミングを行うことができる任意の好適なコンピュータとすることができる。

【００８１】

それぞれの実際のライト１０２は、構成ファイル内に格納されている属性を持つことができる。構成ファイルの構造の一例を図５に示す。複数の実施形態では、構成ファイルは、ライト番号、各ライトの位置、光出力の位置または方向、光のガンマ（明度）、１つまたは複数の属性のインジケータ番号、およびさまざまな他の属性などの各種のデータを収めることができる。構成ファイルに記述されている座標を変更することにより、現実世界のライトをスクリーン上に表現される仮想世界にマッピングし、仮想環境内で生じることを反映するようにできる。したがって、開発者は、爆発などの時間ベースの効果を生成することができる。さまざまなアプリケーション属性に付加することができる効果のライブラリをコード内に用意することができる。例えば、爆発、虹、色の循環的変化（ｃｏｌｏｒ　ｃｈａｓｅ）、フェードイン、フェードアウトなどがある。開発者は、これらの効果をアプリケーション内の仮想オブジェクトに付加する。例えば、爆発を実行した場合、表示装置上で光が消え、構成ファイル内のライトに関連付けられているオブジェクトが破壊されたことを反映する。

【００８２】

構成ファイルを簡素化するために、さまざまな手法を使用することができる。複数の実施形態では、順に並べられた半球状のカメラを倍率設定されたベースラインとして使用して、ライトの三角測量を実行し、そしてライトがどこにあるかを測定しなくても構成ファイルを自動的に発生することができる。複数の実施形態では、構成ファイルをタイプ入力するか、または環境の表現に光源をドラッグ・アンド・ドロップすることができるグラフィカル・ユーザ・インターフェースに入力することができる。開発者は、器具と実際の環境内の本当の配置との照合を行う構成ファイルを作成することができる。例えば、照明構成要素を環境内にドラッグ・アンド・ドロップすると、プログラムはプログラム内の仮想光を環境内の実際の光に関連付けることができる。照明の構成を支援するライト・オーサリング・プログラムの一例が、米国特許出願第０９／６１６２１４号「照明シークエンスをオーサリングするシステム及び方法（Ｓｙｓｔｅｍｓ　ａｎｄ　Ｍｅｔｈｏｄｓ　ｆｏｒ　Ａｕｔｈｏｒｉｎｇ　Ｌｉｇｈｔｉｎｇ　Ｓｅｑｕｅｎｃｅｓ）」で取りあげられている。Ｃｏｌｏｒ　Ｋｉｎｅｔｉｃｓ　Ｉｎｃ．ではさらに、「ＣｏｌｏｒＰｌａｙ」と

いう名称の好適なオーサリングおよび構成プログラムを提供している。

【００８３】

コードの実装に関する詳細については、付録Ａとして付属している「Ｄｉｒｅｃｔｌｉｇｈｔ　ＡＰＩ」文書の中で説明している。Ｄｉｒｅｃｔｌｉｇｈｔ　ＡＰＩは、プログラマが照明効果をプログラムに組み込むためのプログラマ用のインターフェースである。Ｄｉｒｅｃｔｌｉｇｈｔ　ＡＰＩは付録Ａにまとめてあり、開示は参照により本明細書に組み込まれる。オブジェクト指向プログラミングは、照明効果を組み込むために使用されるプログラミング手法の一例にすぎない。照明効果は、任意のプログラミング言語またはプログラミングの方法に組み込むことができる。オブジェクト指向プログラミングでは、プログラマは多くの場合、３Ｄ空間をシミュレートする。

【００８４】

上記の例では、ライトを使用して、予想される光を生成する、または光が付加されるオブジェクトの位置を示していた。光を使用できる方法はほかにも多数ある。ライト・システム内のライトは、コンピュータ・アプリケーション（ゲームなど）で生じる事象を示したり、オブジェクトのレベルまたは属性を示したりするなど、さまざまな目的に使用することができる。

【００８５】

シミュレーション型のコンピュータ・アプリケーションは、多くの場合、３Ｄレンダリングが行われ、オブジェクトは属性並びに事象を備える。プログラマは、現実世界の環境のシミュレーションなどのシミュレーションのため事象をアプリケーションにコーディングすることができる。また、プログラマは、シミュレーションで属性またはオブジェクトをコーディングすることもできる。したがって、プログラムは、爆発、弾丸、価格、製品特徴、健康、他の人々、光のパターンなどの事象および属性を追跡することができる。さらに、コードにより、仮想世界から現実世界にマッピングすることができる。複数の実施形態では、任意選択のステップで、システムはセンサまたは入力デバイスからなどの現実世界のデータを仮想世界に追加することができる。こうして、システムは実世界と仮想世界の対象物を相互に調和する形で制御できる。さらに、ライト・システムをインジケータとして使用することにより、ライト・システムを通じて、現実世界の環境内にいる人を手助けする情報を与えることが可能である。

【００８６】

本明細書では複数の実施形態として、アーキテクチャ面での視覚化、機械工学モデル、およびその他のソリッド・モデリング環境を取りあげている。これらの仮想環境では、照明は多くの場合、仮想環境とソリッド・モデル現実世界視覚化環境の両方に関連する。そこで、ユーザは、ライト・システム１０２の位置決めし且つ制御することにより、現実世界のソリッド・モデルに照明を当て、仮想世界モデリング環境内で生成される照明条件に対応して現実世界のソリッド・モデルに照明を当てることができる。ライトのある部屋の中の物理的スケール・モデルを、１日の間、または１年の間、または異なる季節における照明に関して作成し、場合によっては、ライトおよびさまざまな建物の表面との以前には不明であった相互作用を検出することができる。他の例として、上述のような照明システムを使って室内に都市または都市の一部のレプリカを製作することもできる。そうしてからそのモデルを分析して、一定期間について色の変化がないか、影が生じないか、あるいはその他の照明効果が生じないかを調べることができる。一実施形態では、この手法を景観設計に使用することができる。一実施形態では、照明システムを使用して、部屋、建物、または他の建築部分の内部空間をモデル化する。例えば、インテリア・デザイナは、部屋または部屋の中の構造物や対象物の色を１日、１年、または季節のうちさまざまな時期を表す色で投射することができる。一実施形態では、照明システムをペイント・セクションの近くのストア内で使用し、さまざまな条件の下でペイント色の視覚化のためペイント・チップでの照明条件のシミュレーションを行うことができる。これらの種類の現実世界のモデリング・アプリケーションを利用すると、１年のうちの特定の時期に光反射型建物に反射した日光が運転者の目に入るなどの設計の潜在的欠陥を検知することができる。さ

らに、三次元視覚化を行えば、複雑なコンピュータ・モデリング化よりも人間のほうが設計の美的特質をより素早く認識することができる。

【００８７】

ソリッド・モデリング・プログラムに仮想光を持たせることができる。仮想環境内でモデルを照明することができるが、同時に、現実世界のモデルにも同じようにして照明することができる。例えば、モデルの環境条件をモデル化し、仮想環境の外部にある現実世界のモデリング環境内で作成し直すことができる。例えば、家またはその他の建物をモデル化して、昼の光の環境でどのように見えるかを示すことができる。趣味として楽しむこともでき、その場合は、電車の模型セットの照明をモデル化し（例えば、実際の列車の絵に基づいて）、その照明を電車模型が存在する部屋の照明に変換することもできる。したがって、電車模型は実際の電車の物理的表現というだけでなく、その電車が特定の時期に現れたようにさえ見える。土木工事プロジェクトをモデルとして組み立てることもでき、本発明の原理による照明システムを使用して、１日の特定の期間の照明条件をシミュレートすることもできる。このシミュレーションを使用することにより、照明条件、影、色効果、またはその他の効果を発生することができる。また、この手法を映画館／劇場モデリングで使用したり、映画製作で特殊効果を発生することも可能である。自宅所有者は、例えば、外から見たときの自分の住居の姿形を選択し、そのように見えるようにするライトを選択することにより、このようなシステムを使用することができる。これは、所有者がその場にいないときの安全策の一例である。それとは別に、所有者が家の中のライトをつけ、コンピュータがさまざまな異なる方向および距離から家の様子を示す場合にシステムは逆の動作をすることも可能である。

【００８８】

上記の例では建築のモデリングについて説明しているが、当業者であれば、任意の装置、対象物、または構造についても、その装置、対象物、または構造に当たる光の効果を同様に取り扱うことができる場合に理解するであろう。

【００８９】

医療またはその他の仕事のシミュレーションも実行可能である。本発明の原理による照明システムを使用することにより、医療処置が行われているときの照明条件をシミュレートすることができる。この作業では、特定の照明条件により夜間交通事故などの手術室のセッティングを行うか、またはその他の環境を生成する必要がある。例えば、高速道路の照明は一般に高圧ナトリウム灯であり、それは黄色に近い単色光を発生するため、物体や流体はふつうと違う色に見えることがある。駐車場では、一般に、金属ハロゲン化物照明システムを使用しており、広域スペクトル光を発生し、スペクトルのギャップがある。これらの環境は、いずれも、本発明の原理によるシステムを使用してシミュレートすることができる。シミュレータを使用することにより、異なる照明下の状況でどのように反応すべきかについて救急隊員を訓練することができる。さらに、作業を行う際に課される条件のシミュレーションにも使用できる。例えば、周回軌道衛星の修理を行う宇宙飛行士を照明する光を地上のシミュレーション室でシミュレートすることができる。

【００９０】

またライトを使用することで、空間内を進行するまたは天文学的現象を見せる光が届くなど他の方法ではアクセスできない範囲内の進行をシミュレートしたり、あるいはライトを他の方法では表示することができない物体の三次元投影として使用することもできる。例えば、コンピューティング・デバイスに接続されている照明システムは、分子モデルの内部から三次元表示を行うことが可能である。時間的関数またはその他の数学的概念もまた、視覚化することが可能である。

【００９１】

上で述べたすべての記事、特許、およびその他の参考文献は参照により本明細書に組み込まれる。本発明は、図示され、詳細に説明されている実施形態に関して開示されているが、さまざまな等価物、修正、および改良は、当業者であれば上の説明から明白であろう。

【００９２】
・付録Ａ
最初に読まなければならない重要事項。
【００９３】
１）サンプルプログラムおよびＲｅａｌ　Ｌｉｇｈｔ　Ｓｅｔｕｐは、ＤｉｒｅｃｔＬｉｇｈｔ．ｄｌｌ　ＣＯＭオブジェクトを、使用しているコンピュータ上のＷｉｎｄｏｗｓに登録してからでないと起動しない。「ＲｅｇｉｓｔｅｒＤｉｒｅｃｔＬｉｇｈｔ．ｅｘｅ」および「ＵｎｒｅｇｉｓｔｅｒＤｉｒｅｃｔＬｉｇｈｔ．ｅｘｅ」とうまく名付けられた２つの小さなプログラムが、このインストール・プログラムに付属している。
【００９４】
２）ＤｉｒｅｃｔＬｉｇｈｔでは、ＳｍａｒｔＪａｃｋがＣＯＭ１に割り当てられている（ｈｏｏｋｅｄ　ｕｐ　ｔｏ）ことを想定している。この想定を変更するには、ファイル「ｍｙ＿ｌｉｇｈｔｓ．ｈ」内のＤＭＸ＿ＩＮＴＥＲＦＡＣＥ＿ＮＵＭの値を編集する。
【００９５】
ＤｉｒｅｃｔＬｉｇｈｔについて
組織
アプリケーション（例えば、３Ｄ表示ゲーム）は、その３Ｄ世界の中に仮想光を生成することができる。ＤｉｒｅｃｔＬｉｇｈｔによって、これらの光が現実世界のＣｏｌｏｒ　Ｋｉｎｅｔｉｃｓ全スペクトルディジタル光にマッピングされ、色および明度の設定はゲーム内の仮想光の位置と色に対応する。
【００９６】
ＤｉｒｅｃｔＬｉｇｈｔｓでは、以下の３つの一般的な種類の仮想光が存在する。
動的光。最も一般的な形式の仮想光は位置と色の値を持つ。この光は、必要なだけ、移動とその色の変更を繰り返すことができる。動的光は、光を放つ星雲、ロケットから吹き出される炎、企業のロゴを照らして過ぎてゆく黄色のスポットライト、または飢えた突然変異のアイスいたち（ｉｃｅ－ｗｅａｓｅｌ）の赤く輝く目を表現することができる。
【００９７】
周囲光は静止しており、色値のみを持つ。日光、頭上ルーム・ライト、または一般的なカラー・ウォッシュは、周囲光の例である。動的光およびインジケータ光は好きなだけ設定できるが、周囲光源は１つしか設定できない（これは結局周囲色値となる）。
【００９８】
インジケータ光は、特定の現実世界の光にしか割り当てられない。動的光は位置を変えることができ、したがって異なる現実世界の光に影響を及ぼすが、周囲光はあらゆる現実世界の光に影響を及ぼす一定の色であり、インジケータ光は常に、単一の現実世界の光にのみ影響する。インジケータは、例えば、シールド状態、スレッド配置などの照明と別のフィードバックをユーザに与えることを意図している。
これらすべての光により必要な回数だけ色を変更することができる。
【００９９】
一般に、ユーザは現実世界の光をセットアップする。「ｍｙ＿ｌｉｇｈｔｓ．ｈ」構成ファイルは、「ＤｉｒｅｃｔＬｉｇｈｔ　ＧＵＩ　Ｓｅｔｕｐ」プログラム内に作成され、それにより編集することができる。ＡＰＩにより、現実世界のライトがどこにあり、どのような種類であるか、およびどの種類の仮想光（動的光、周囲光、インジケータ、または何らかの組み合わせ）が影響を及ぼすかについての情報すべてを含む「ｍｙ＿ｌｉｇｈｔｓ．ｈ」ファイルから設定がロードされる。
【０１００】
仮想光を作成し、静的な光とするか、または実行時に動的に作成することができる。ＤｉｒｅｃｔＬｉｇｈｔｓは、それ専用のスレッドで実行され、常時、新しい値をライト内にポーク（ｐｏｒｋ）し、眠りに落ちないようにする。仮想光を更新した後、１回関数を呼び出してそれを現実世界の光に送る。ＤｉｒｅｃｔＬｉｇｈｔｓでは、仮想世界から現

実世界へのすべてのマッピングを処理する。

【０１０１】

アプリケーション側ですでに３Ｄ光源を使用している場合、光源は１対１でＶｉｒｔｕａｌ＿Ｌｉｇｈｔクラスにマッピングにされるため、ＤｉｒｅｃｔＬｉｇｈｔの実装は非常に簡単である。

【０１０２】

アクション・ゲームの通常セットアップでは、１つのオーバーヘッド・ライトは主に周囲光に設定され、モニタの後ろ、側面、および周りのライトは主に動的光に設定され、スクリーン付近のたぶんいくつかの小さなライトはインジケータに設定される。

【０１０３】

周囲光により、雰囲気と趣が作り出される。プレーヤの回りの動的光は、武器、環境オ対象物、爆発音など、プレーヤの回りで生じていることに関するフィードバックを与える。インジケータ・ライトは、シールドレベル、危険、検出などの、ゲーム・パラメータに関する瞬間的フィードバックを返す。

【０１０４】

効果（ＬｉｇｈｔｉｎｇＦＸ）を、動的照明をオーバライドまたは増強するライトに付加することができる。例えば、Ｓｔａｒ　Ｔｒｅｋ：Ａｒｍａｄａでは、Ｒｅｄ　Ａｌａｒｔを叩くと、室内のすべてのライトが赤色のパルスを発し、一時的にそのライトが持つ他の色情報を置き換える。

【０１０５】

別の効果で増強することができる。例えば、爆発効果を単一の仮想光に付加して、長時間演出させれば、火の玉を次第に消してゆくために値を絶えず微調整しなくても、仮想光を作成し、効果を付加して起動することができ、効果が完了するまでライトをそのままにしておくことができる。

【０１０６】

実際のライトには、ライトが設置されている部屋に基づく座標系が設定されている。コンピュータのモニタの前に座っている人を基準とし、その頭を原点とみなす。Ｘは右方向に増大する。Ｙは天井方向に増大する。Ｚはモニタ方向に増大する。

【０１０７】

仮想光では全く自由にどのような座標系をも使用することができる。仮想光を実際の光にマッピングする異なるいくつかのモードがある。仮想光の座標系の軸を実際の光の座標系に揃えることにより、かなり楽に行えるようになる。

【０１０８】

ライト位置は任意の実際の値をとることができる。ＤｉｒｅｃｔＬｉｇｈｔ　ＧＵＩセットアップ・プログラムでは、ライトの位置を室内の中心から１ｍの範囲内に制限しているが、必要なら手作業で自分が思っている内容に値を変更することもできる。ただし、まず最初に投射タイプについてよく読んでいただきたい。いくつかのモードでは、現実世界および仮想世界の座標系が同じ尺度を持つ必要がある。

【０１０９】

はじめに

ＤｉｒｅｃｔＬｉｇｈｔ　ＳＤＫのインストール方法

Ｓｅｔｕｐ．ｅｘｅファイルを実行するとインストールされる内容：

／Ｗｉｎｄｏｗｓ／Ｓｙｓｔｅｍ／に、３つのｄｌｌファイル、つまりＤｉｒｅｃｔＬｉｇｈｔ用のファイル１つと、ＤＭＸを介した現実世界の光との低レベル通信用のファイル２つがインストールされる。

【０１１０】

DirectLight/dll
DMXIO.dll
DLPORTIO.dll

【０１１１】

ＤｉｒｅｃｔＬｉｇｈｔがインストールされているフォルダ内に置かれるのは、以下のＶｉｓｕａｌ　Ｃ＋＋プロジェクト・ファイル、ソース・コード、およびヘッダ・ファイル：

DirectLight.dsp
DirectLight.dsw

など。

DirectLight.h
DirectLight.cpp
Real_Light.h
Real_Light.cpp
Virtual_Light.h
Virtual_Light.cpp

など。

【０１１２】

コンパイル時ライブラリ：

FX_Library.lib
DirectLight.lib
DMXIO.lib

および、構成ファイル：

my_lights.h
light_definitions.h
GUI_config_file.h
Dynamic_Localized_Strings.h

【０１１３】

「ｍｙ＿ｌｉｇｈｔｓ．ｈ」ファイルは、ＤｉｒｅｃｔＬｉｇｈｔとＤｉｒｅｃｔＬｉｇｈｔ　ＧＵＩ　Ｓｅｔｕｐ．ｅｘｅの両方から参照される。さらに、「ｍｙ＿ｌｉｇｈｔｓ．ｈ」は　「ｌｉｇｈｔ＿ｄｅｆｉｎｉｔｉｏｎｓ．ｈ」を参照する。他のファイルは、ＤｉｒｅｃｔＬｉｇｈｔ　ＧＵＩ　Ｓｅｔｕｐのみが参照する。ＤＬＬとＳｅｔｕｐプログラムは両方とも、これらのファイルを見つけるためにレジストリ・エントリを使用する。

【０１１４】

【表１】

HKEY_LOCAL_MACHINE\Software\ColorKinetics\DirectLight\1.00.000\location

【０１１５】

さらに、このディレクトリには、以下のドキュメントおよびサブフォルダが含まれる。

ＦＸ＿Ｌｉｂｒａｒｉｅｓには、ＤｉｒｅｃｔＬｉｇｈｔｓによってアクセスできる照明効果が格納されている。

Ｒｅａｌ　Ｌｉｇｈｔ　Ｓｅｔｕｐには、実際のライトに関する情報を変更するためのグラフィカル・エディタが格納されている。

Ｓａｍｐｌｅ　Ｐｒｏｇｒａｍには、ＤｉｒｅｃｔＬｉｇｈｔの使用法を説明する注釈が多数記述されたプログラムが格納されている。

【０１１６】

ＤｉｒｃｅｔＬｉｇｈｔ　ＣＯＭ

ＤｉｒｅｃｔＬｉｇｈｔ　ＤＬＬは、ＤｉｒｅｃｔＬｉｇｈｔ機能をカプセル化したＣＯＭオブジェクトを実装する。ＤｉｒｅｃｔＬｉｇｈｔオブジェクトは、クライアント・プログラムによって使用されるＤｉｒｅｃｔＬｉｇｈｔインターフェースを所有する。

【０１１７】

ＤｉｒｅｃｔＬｉｇｈｔ　ＣＯＭオブジェクトを使用するには、そのオブジェクトが使用されるマシンに、ＤｉｒｅｃｔＬｉｇｈｔ　ＣＯＭサーバを登録していなければならな

い（上の「最初に読まなければならない重要事項」を参照）。この作業を行っていない場合、Ｍｉｃｒｏｓｏｆｔ　ＣＯＭランタイム・ライブラリはＣＯＭサーバが置かれている場所を認識しない（本質的に、ＤｉｒｅｃｔＬｉｇｈｔ．ｄｌｌのパスが必要である）。

【０１１８】

プログラム（クライアントと呼ぶ）からＤｉｒｅｃｔＬｉｇｈｔ　ＣＯＭオブジェクトにアクセスするには、まず、ＤｉｒｅｃｔＬｉｇｈｔ　ＣＯＭインターフェース（とりわけ）の定義を含む「ｄｉｒｅｃｔｌｉｇｈｔ．ｈ」と、オブジェクトおよびインターフェースのさまざまなＵＩＤの定義を含む「ｄｉｒｅｃｔｌｉｇｈｔ＿ｉ．ｃ」を含まなければならない（後で詳しく説明する）。

【０１１９】

ＣＯＭサービスを使用する前に、まず、ＣＯＭランタイムを初期化する必要がある。そのために、ＮＵＬＬパラメータでもってＣｏＩｎｉｔｉａｌｉｚｅ関数を呼び出す。

```
CoInitialize(NULL)
```

この場合、戻り値は気にしなくてよい。

【０１２０】

次に、ＤｉｒｅｃｔＬｉｇｈｔオブジェクトをインスタンス化しなければならない。そのためには、ＣｏＣｒｅａｔｅＩｎｓｔａｎｃｅ関数を呼び出す必要がある。これにより、ＤｉｒｅｃｔＬｉｇｈｔオブジェクトのインスタンスが作成され、ＤｉｒｅｃｔＬｉｇｈｔインターフェースへのポインタが返される。

【０１２１】

【表２】

```
HRESULT hCOMError =
   CoCreateInstance( CLSID_CDirectLight,
                     NULL,
                     CLSCTX_ALL ,
                     IID_IDirectLight,
                     (void **)&pDirectLight);
```

【０１２２】

ＣＬＳＩＤ＿ＣＤｉｒｅｃｔＬｉｇｈｔは、ＤｉｒｅｃｔＬｉｇｈｔオブジェクトの識別子（ｄｉｒｅｃｔｌｉｇｈｔ＿ｉ．ｃで宣言されている）であり、ＩＩＤ＿ＤｉｒｅｃｔＬｉｇｈｔは、ＤｉｒｅｃｔＬｉｇｈｔインターフェースの識別子であり、ｐＤｉｒｅｃｔＬｉｇｈｔは、ここでインスタンス化したばかりのオブジェクト上のＤｉｒｅｃｔＬｉｇｈｔインターフェースの実装へのポインタである。

【０１２３】

ＣｏＣｒｅａｔｅＩｎｓｔａｎｃｅによってエラーが返された場合、エラーＲＥＧＤＢ＿Ｅ＿ＣＬＡＳＳＮＯＴＲＥＧである可能性が最も高く、クラスが使用されているマシンに登録されていないことを示す。その場合、Ｒｅｇｉｓｔｅｒ　ＤｉｒｅｃｔＬｉｇｈｔプログラムを実行して、再び試みる必要がある。

【０１２４】

アプリケーションをクリーンアップする場合、以下の３行を書き込まなければならない。

```
// ＣＯＭオブジェクトを終了する。
pDirectLight->Release();
// ＣＯＭに、未使用ＣＯＭサーバのアンロードを依頼する。
CoFreeUnusedLibraries();
// このアプリケーションを終了し、ＣＯＭライブラリをシャットダウンする。
CoUninialize();
```

【０１２５】

これを使用して実行する場合には、ＣＯＭインターフェースを必ず解放しなければならない。そうしないと、アプリケーションを終了した後もオブジェクトがメモリ内に残るこ

とになる。

【０１２６】

ＣｏＦｒｅｅＵｎｕｓｅｄＬｉｂｒａｒｉｅｓ（）は、ＤｉｒｅｃｔＬｉｇｈｔファクトリ（ＣｏＣｒｅａｔｅＩｎｓｔａｎｃｅ（）を呼び出すときにＣＯＭオブジェクトを作成したサーバ）をメモリから削除するようにＣＯＭに依頼し、ＣｏＵｎｉｎｉｔｉａｌｉｚｅ（）はＣＯＭライブラリをシャットダウンする。

【０１２７】

ＤｉｒｅｃｔＬｉｇｈｔクラス

ＤｉｒｅｃｔＬｉｇｈｔクラスは、ＡＰＩのコア機能を含む。周囲光値、すべてのライトの大域的明度（ガンマ）の設定、および仮想光の追加および削除を行う機能が含まれる。

種類：

【０１２８】

【表３】

```
enum Projection_Type{
      SCALE_BY_VIRTUAL_DISTANCE_TO_CAMERA_ONLY =0,
      SCALE_BY_DISTANCE_AND_ANGLE = 1,
      SCALE_BY_DISTANCE_VIRTUAL_TO_REAL = 2 };
```

【０１２９】

これらの値の詳細については、Ｄｉｒｅｃｔ　Ｌｉｇｈｔ　Ｃｌａｓｓの中の「Ｐｒｏｊｅｃｔｉｏｎ　Ｔｙｐｅｓ」を参照のこと。

【０１３０】

【表４】

```
enum Light_Type{
      C_75 = 0,
      COVE_6 = 1 };
```

【０１３１】

これらの値の詳細については、Ｄｉｒｅｃｔ　Ｌｉｇｈｔ　Ｃｌａｓｓの「Ｌｉｇｈｔ　Ｔｙｐｅｓ」を参照するか、または「ＤｉｒｅｃｔＬｉｇｈｔ　ＧＵＩ　Ｓｅｔｕｐ」のオンラインヘルプを参照のこと。

【０１３２】

【表５】

```
enum Curve_Type{
      DIRECTLIGHT_LINEAR = 0,
      DIRECTLIGHT_EXPONENTIAL = 1,
      DIRECTLIGHT_LOGARITHMIC = 2 };
```

【０１３３】

これらの値は、一つの色から他の色へフェードするときの照明効果について異なる曲線を示している。

パブリックメンバ関数：

【０１３４】

【表６】

```
void Set_Ambient_Light( int R,
                        int G,
                        int B   );
```

【０１３５】

Ｓｅｔ＿Ａｍｂｉｅｎｔ＿Ｌｉｇｈｔ関数は、周囲光の赤色、緑色、および青色の値を、関数に渡す値に設定する。これらの値は、０～ＭＡＸ＿ＬＩＧＨＴ＿ＢＲＩＧＨＴＮＥＳＳの範囲である。周囲光は、アプリケーションにおいて定数または「ルームライト」を表すように設計されている。周囲光は、現実世界のライトのうちどれかのライトまたはす

べての現実のライトに送ることができる。それぞれの現実世界のライトは、周囲光のうちの任意の割合の周囲光を含むことができる。

【０１３６】

void Stir_Lights(void * user_data,);

Ｓｔｉｒ＿Ｌｉｇｈｔｓは、ＤｉｒｅｃｔＬｉｇｈｔｓ内に作成されているライトバッファに基づいて現実世界のライトにライト情報を送る。ＤｉｒｅｃｔＬｉｇｈｔ　ＤＬＬは、自動的に光のかき回しを処理する。この関数は、通常、アプリケーションからは呼ばれない。

【０１３７】

【表７】

```
Virtual_Light * Submit_Virtual_Light( float xpos,
                                      float ypos,
                                      float zpos,
                                      int red,
                                      int green,
                                      int blue   );
```

【０１３８】

Ｓｕｂｍｉｔ＿Ｖｉｒｔｕａｌ＿Ｌｉｇｈｔは、Ｖｉｒｔｕａｌ＿Ｌｉｇｈｔインスタンスを作成する。その仮想位置は、渡された最初の３つの値で指定し、その色は、残りの３つの値で指定する。位置については、アプリケーション空間座標を使用しなければならない。色の値は、０～ＭＡＸ＿ＬＩＧＨＴ＿ＢＲＩＧＨＴＮＥＳＳの範囲である。この関数は、作成されたライトへのポインタを返す。

【０１３９】

void Remove_Virtual_Light(Virtual_Light * bad_light);

Ｖｉｒｔｕａｌ＿Ｌｉｇｈｔインスタンスへのポインタが与えられると、Ｒｅｍｏｖｅ＿Ｖｉｒｔｕａｌ＿Ｌｉｇｈｔにより仮想光が削除される。

【０１４０】

void Set_Gamma(float gamma);

Ｓｅｔ＿Ｇａｍｍａ関数は、Ｄｉｒｅｃｔ　Ｌｉｇｈｔデータ構造体のガンマ値を設定する。すべての仮想光は実際のライト上に投射される前にガンマ値を乗算されるので、すべての光の値を全体として制御する場合にこの値を使用することができる。

【０１４１】

void Set_Cutoff_Range(float cutoff_range);

Ｓｅｔ＿Ｃｕｔｏｆｆ＿Ｒａｎｇｅは、カメラからのカットオフ距離を設定する。仮想光は、この距離を超えると、現実世界のライトに影響しなくなる。値を高い値に設定して、仮想光が遠く離れたところから現実世界のライトに影響を及ぼすようにする。この値が小さい場合には、効果を生じさせるために仮想光をカメラに近づける必要がある。値は、アプリケーション空間座標で指定しなければならない。

【０１４２】

void Clear_All_Real_Lights(void);

Ｃｌｅａｒ＿Ａｌｌ＿Ｌｉｇｈｔｓはすべての実際のライトを破壊する。

【０１４３】

void Project_All_Lights(void);

Ｐｒｏｊｅｃｔ＿Ａｌｌ＿Ｌｉｇｈｔｓは、現実世界のすべてのライトに対するすべての効果を計算するが、その際に、ガンマ、周囲光、および動的光の関与、位置および投射モード、カットオフ角度およびカットオフ範囲を考慮し、値をすべての現実世界のライトに送る。

【０１４４】

【表８】

```
void Set_Indicator_Color( int which_indicator,
                          int red,
                          int green,
                          int blue   );
```

【０１４５】

構成ファイル（ｍｙ＿ｌｉｇｈｔｓ．ｈ）を介して現実世界のライトにインジケータを割り当てることができる。各インジケータには、一意的な非負整数ＩＤが設定されていなければならない。Ｓｅｔ＿Ｉｎｄｉｃａｔｏｒ＿Ｃｏｌｏｒは、ｗｈｉｃｈ＿ｉｎｄｉｃａｔｏｒによって指定されているインジケータの色を指定されている赤色、緑色、および青色の値に変更する。存在していないインジケータＩＤでＳｅｔ＿Ｉｎｄｉｃａｔｏｒ＿Ｃｏｌｏｒを呼び出しても何も実行されない。ユーザ側で、どのライトをインジケータにすべきかを指定するが、それでもインジケータであるライトは周囲光および動的光の影響を受ける。

【０１４６】

Indicator Get_Indicator(int which_indicator);

指定された値を持つインジケータへのポインタを返す。

int Get_Real_Light_Count(void);

実際のライトの個数を返す。

【０１４７】

void Get_My_Lights_Location(char buffer(MAX_PATH)

ディレクトリ内を探索し「ｍｙ＿ｌｉｇｈｔｓ．ｈ」ファイルへのパスを見つける。

void Load_Real_Light_Configuration(char * fullpath = NULL);

「ｍｙ＿ｌｉｇｈｔｓ．ｈ」ファイルをレジストリで決められているデフォルトの場所からロードする。ＤｉｒｅｃｔＬｉｇｈｔは、このファイル内の情報に基づいて実際のライトのリストを作成する。

【０１４８】

【表９】

```
void Submit_Real_Light( char * indentifier,
                        int DMX_port,
                        Projection_Type projection_type,
                        int indicator_number,
                        float add_ambient,
                        float add_dynamic,
                        float gamma,
                        float cutoff_angle,
                        float x,
                        float y,
                        float z   );
```

【０１４９】

現実世界に新しい実際のライトを作成する。通常、ＤｉｒｅｃｔＬｉｇｈｔは、起動時に、「ｍｙ＿ｌｉｇｈｔｓ．ｈ」ファイルから実際のライト情報をロードする。

void Remove_Real_Light(Real_Light * dead_light);

実際のライトのインスタンスを安全に削除する。

【０１５０】

Light GetAmbientLight (void);

周囲光へのポインタを返す。

bool RealLightListEmpty (void);

実際のライトのリストが空であれば真を返し、そうでなければ偽を返す。

【０１５１】

Ｌｉｇｈｔクラス

周囲光はライトとして定義されている。Ｌｉｇｈｔクラスは、Ｖｉｒｔｕａｌ　Ｌｉｇｈｔｓと Ｒｅａｌ　Ｌｉｇｈｔｓの親クラスである。メンバ変数：

static const int Max_LIGHT_BRIGHTNESS. ２５５と定義されている。

【０１５２】

LightingFX_List * m_FX_currently_attached. このライトに現在割り当てられている効果のリスト。

ColorRGB m_color. ライトはすべて、色が設定されていなければならない。ＣｏｌｏｒＲＧＢはＣｏｌｏｒＲＧＢ．ｈで定義されている。

【０１５３】

void Attach_FX(LightingFX * new_FX)
新しい照明効果をこの仮想光に付加する。

void Detach_FX(LightingFX * old_FX)
古い照明効果をこの仮想光から削除する。

【０１５４】

Ｒｅａｌ Ｌｉｇｈｔｓ

Ｒｅａｌ Ｌｉｇｈｔは、Ｌｉｇｈｔクラスからの継承である。実際のライトは、仮想世界内のライトを表す。メンバ変数：

static const int NOT_AK_INDICATOR_LIGHT. －１と定義されている。

【０１５５】

ｃｈａｒ ｍ＿ｉｄｅｎｔｉｆｉｅｒ［１００］は、ライトの名前である（例えば、「ｏｖｅｒｈｅａｄ」や「ｃｏｖｅｌｉｇｈｔ１」など）。デバッグツールとして以外は、ＤｉｒｅｃｔＬｉｇｈｔによって使用されない。

【０１５６】

ｉｎｔ ＤＭＸ＿ｐｏｒｔは、チャネルを表す一意的な非負整数である。指定されたライトは情報を受け取る。ＤＭＸ情報は、ライト毎に３バイトずつ（赤色、緑色、および青色）バッファ内に送り出される。（ＤＭＸ＿ｐｏｒｔ＊３）は、実際には、指定されたライトに対する赤色値のインデックスである。ＤｉｒｅｃｔＬｉｇｈｔ ＤＭＸバッファは５１２バイトなので、ＤｉｒｅｃｔＬｉｇｈｔは約１７０個のライトをサポートすることができる。バッファが大きいとパフォーマンスに問題が生じる可能性があるため、できれば、ＤＭＸ＿ｐｏｒｔに大きな数を使用するのを避ける必要がある。

【０１５７】

Ｌｉｇｈｔ＿Ｔｙｐｅ ｍ＿ｔｙｐｅは、Ｃｏｌｏｒ Ｋｉｎｅｔｉｃｓライトの異なるモデルを示す。今のところ、ＤｉｒｅｃｔＬｉｇｈｔ ＧＵＩ Ｓｅｔｕｐでアイコンを表示する以外は未使用である。

【０１５８】

ｆｌｏａｔ ｍ＿ａｄｄ＿ａｍｂｉｅｎｔは、このライトの色に対する周囲光の寄与の程度を示す。範囲は０～１である。

ｆｌｏａｔ ｍ＿ａｄｄ＿ｄｙｎａｍｉｃは、このライトの色に対する動的光の寄与の程度を示す。範囲は０～１である。

【０１５９】

ｆｌｏａｔ ｍ＿ｇａｍｍａは、このライトの全体的な明度である。範囲は０～１である。

ｆｌｏａｔ ｍ＿ｃｕｔｏｆｆ＿ａｎｇｌｅは、ライトが周囲の仮想光の寄与にどの程度敏感かを決定する。値が大きいと、ほとんどの仮想光から情報を受け取る。値が小さいと、実際のライトと同じアークの仮想光からの寄与分のみを受け取る。

【０１６０】

Ｐｒｏｊｅｃｔｉｏｎ＿Ｔｙｐｅ ｍ＿ｐｒｏｊｅｃｔｉｏｎ＿ｔｙｐｅは、仮想光を実際のライトにどのようにマッピングするかを定義する。

ＳＣＡＬＥ＿ＢＹ＿ＶＩＲＴＵＡＬ＿ＤＩＳＴＡＮＣＥ＿ＴＯ＿ＣＡＭＥＲＡ＿ＯＮＬＹ。この実際のライトは、仮想座標系の原点から仮想光の位置までの距離にのみ基づき仮想光からの寄与分を受け取る。仮想光の寄与分は、原点からの距離がカットオフ範囲に近づくにつれ直線的に減少する。

【０１６１】
ＳＣＡＬＥ＿ＢＹ＿ＤＩＳＴＡＮＣＥ＿ＡＮＤ＿ＡＮＧＬＥ。この実際のライトは、上で計算した距離および実際の光と仮想光との間の角度の差に基づいて仮想光からの寄与分を受け取る。仮想光の寄与分は、原点からの距離がカットオフ範囲に近づき、角度がカットオフ角度に近づくにつれ直線的に減少する。
【０１６２】
ＳＣＡＬＥ＿ＢＹ＿ＤＩＳＴＡＮＣＥ＿ＶＩＲＴＵＡＬ＿ＴＯ＿ＲＥＡＬ。この実際のライトは、実際の光から仮想光までの３次元空間内での距離に基づいて仮想光からの寄与分を受け取る。このモードでは、実際の座標系と仮想座標系とが同じであると仮定している。仮想光寄与分は、実際の光から仮想光までの距離がカットオフ範囲に近づくにつれ直線的に減少する。
【０１６３】
ｆｌｏａｔ　ｍ＿ｘｐｏｓ　　仮想空間内のｘ，ｙ，ｚ位置。
ｆｌｏａｔ　ｍ＿ｙｐｏｓ
ｆｌｏａｔ　ｍ＿ｚｐｏｓ
インジケータが負の場合、ライトはインジケータではない。非負の場合、そのインジケータ番号に送られた色のみを受け取る。
【０１６４】
Ｖｉｒｔｕａｌ　Ｌｉｇｈｔｓ
Ｖｉｒｔｕａｌ　Ｌｉｇｈｔｓは、現実世界のＣｏｌｏｒ　Ｋｉｎｅｔｉｃｓライトにマッピングされるゲームまたはその他のリアルタイムアプリケーション内の光源を表す。Ｖｉｒｔｕａｌ　Ｌｉｇｈｔｓは、作成、移動、破壊が可能で、その色は、アプリケーション内で可能な限り何回でも変更することができる。
【０１６５】
```
static const int MAX_LIGHT_BRIGHTNESS;
```
ＭＡＸ＿ＬＩＧＨＴ＿ＢＲＩＧＨＴＮＥＳＳは、ライトに設定可能な最大の値を表す定数である。ほとんどのＣｏｌｏｒ　Ｋｉｎｅｔｉｃｓライトの場合、この値は２５５である。ライトは、０から始まるある範囲を持つと仮定される。
【０１６６】
```
void Set_Color( int R,
                int G,
                int B );
```
Ｓｅｔ＿Ｃｏｌｏｒ関数は、仮想光の赤色、緑色、および青色の値を、関数に渡す値に設定する。
【０１６７】
```
void Set_Position( float x_pos,
                   float y_pos,
                   float z_pos );
```
Ｓｅｔ＿Ｐｏｓｉｔｉｏｎ関数は、仮想光の位置の値を、関数に渡す値に設定する。位置については、アプリケーション空間座標を使用しなければならない。
【０１６８】
```
void Get_Position( float *x_pos,
                   float *y_pos,
                   float *z_pos );
```
ライトの位置を獲得する。
【０１６９】
Ｌｉｇｈｔｉｎｇ　ＦＸ
Ｌｉｇｈｔｉｎｇ　ＦＸは、実際のライトまたは仮想光、またはインジケータ、さらには周囲光にさえも付加することができる時間ベースの効果である。照明効果には、子としての別の効果もあり、その場合、子は順次再生される。

【０１７０】
static const int FX_OFF; －１と定義されている。
static const int START_TIME;
static const int STOP_TIME:
効果を開始および停止する時間。これは仮想の値である。個々の効果は、全体に基づいて再生の時間をスケーリングする。
【０１７１】
void Set_Real_Time(bool Real_Time);
ＴＲＵＥを渡した場合、この効果は、現実世界の時間が使用され、Ｓｔｉｒ＿Ｌｉｇｈｔｓが呼び出される回数だけ自動的に更新される。ＦＡＬＳＥ（偽）が渡されると、効果ではアプリケーション時間が使用され、Ａｐｐｌｙ＿ＦＸが呼び出される毎に更新される。
【０１７２】
void Set_Time_Extrapolation (bool extrapolate);
ＴＲＵＥ（真）が渡されると、この効果では、Ｓｔｉｒ＿Ｌｉｇｈｔｓが呼び出されたときにその値を外挿する。
【０１７３】
void Attach_FX_To_Light (Light * the_Light);
この効果を渡されたライトに付加する。
void Detach_FX From_Light (Light * the_light;
bool remove_FX_from_light = true);
ライトに対するこの効果の寄与分を削除する。ｒｅｍｏｖｅ＿ＦＸ＿ｆｒｏｍ＿Ｌｉｇｈｔが真の場合、効果はさらに、ライトからも削除される。
【０１７４】
上記の関数はさらに、仮想光、インジケータ・ライト（インジケータへのポインタまたはその番号のいずれかによって参照される）、周囲光、およびすべての実際のライトを有効にするバージョンとして存在する。
【０１７５】
void Start (float FX_play_time,
bool looping = false);
効果を開始する。ｌｏｏｐｉｎｇが真であれば、効果は終了した後再び開始する。
void Stop (void);
破壊せずに効果を停止する。
【０１７６】
void Time_Is_Up (void);
時間切れになってから、効果の再生をループするかまたは停止する。
void Update_Time (float time_passed);
この効果に関して経過したゲーム時間を変更する。
【０１７７】
void Update_Real_Time (void);
この効果に関して経過した実時間を調べる。
void Update_Extrapolated_Time (void);
リアルタイムでこれまでに経過したアプリケーション時間の外挿に基づいてＦＸ時間を変更する。
【０１７８】
virtual void Apply_FX (ColorRGB base_color);
これは、主照明関数である。Ｌｉｇｈｔｉｎｇ ＦＸが継承されている場合、この関数は、時間の経過とともにライトの色値を実際に変更するすべての重要な作業を実行する。既存のライト値に値を追加するか、既存の値を別の値で置き換えるか、またはその２つの組み合わせを行うことを選択できることに注意されたい。このようにして、Ｌｉｇｈｔｉ

ｎｇ効果は既存のライトを指定変更したり、単に取って代わることができる。
【０１７９】
　static void Update_All_FX_Time (float time_passed);
すべての効果の時間を更新する。
　void Apply_FX_To_All_Virtual_Lights (void);
この効果を、適切なすべての仮想光、周囲光、およびインジケータ・ライトに適用する。
【０１８０】
　void Apply_ALL_FX_To_All_Virtual_Lights (void);
各効果を、適切なすべての仮想光、周囲光、およびインジケータ・ライトに適用する。
　void Apply_ALL_FX_To_Real_Light (Real_Light * the_real_light);
この効果を単一の実際のライトに適用する。
【０１８１】
　void Start_Next_ChildFX (void);
この効果が子効果を持つ場合、次の操作を開始する。
　void Add_ChildFX (LightingFX * the_child,
　　float timeshare);
新しい子効果を、この効果が持つ子効果のリストの末尾に追加する。ｔｉｍｅｓｈａｒｅは、効果が再生される総時間に対する子の時間の割合である。割り当て分合計が効果の実際の再生時間合計に一致するようにスケーリングされるため、ｔｉｍｅｓｈａｒｅは合計して１になる必要はない。
【０１８２】
　void Become_Child_Of (Lighting_FX * the_parent);
指定された効果の親になる。
　void Inherit_Light_List (Affected_Lights * our_lights);
この効果を所有し、その子すべてが影響のあるライトのリストを継承する。
【０１８３】
構成ファイル
ファイル「ｍｙ＿ｌｉｇｈｔｓ．ｈ」には、現実世界のライトに関する情報が格納され、起動時にＤｉｒｅｃｔＬｉｇｈｔシステムにロードされる。ファイル「ｍｙ＿ｌｉｇｈｔｓ．ｈ」および「ｌｉｇｈｔ＿ｄｅｆｉｎｉｔｉｏｎｓ．ｈ」は、ＤｉｒｅｃｔＬｉｇｈｔｓを使用するアプリケーションと同じディレクトリに格納されていなければならない。
【０１８４】
「ｍｙ＿ｌｉｇｈｔｓ．ｈ」は、ＤｉｒｅｃｔＬｉｇｈｔ　ＧＵＩ　Ｓｅｔｕｐプログラムによって作成され、編集される。このプログラムの使用法については、プログラムのオンラインヘルプを参照されたい。
「ｍｙ＿ｌｉｇｈｔｓ．ｈ」ファイルの例を以下に示す。
【０１８５】

【表１０】

```
//////////////////////////////////////////////////////////////
//
// my_lights.h
//
// Configuration file for Color Kinetics lights
//          used by DirectLights                    (注１)
//
// This file created with DirectLights GUI Setup v1.0
//
//////////////////////////////////////////////////////////////

// Load up the basic structures (注２)
#include "Light_Definitions.h"

// overall gamma (注３)
float OVERALL_GAMMA = 1.0;

// which DMX interface do we use? (注４)
int DMX_INTERFACE_NUM = 0;

//////////////////////////////////////////////////////////////
//
// This is a list of all the real lights in the world (注５)
//
Real_Light my_lights[MAX_LIGHTS] =
{
//NAME          PORT  TYPE PRJ IND   AMB     DYN     GAMMA   CUTOFF     X        Y        Z
"Overhead",      0,    1,  0, -1,  1.000,  0.400,  1.000,  3.142,  0.000,  -1.000,  0.000,
"Left",          1,    0,  1, -1,  0.000,  1.000,  1.000,  1.680,  -1.000,  0.000,  0.000,
"Right",         2,    0,  1, -1,  0.000,  1.000,  0.800,  1.680,  1.000,  0.000,  0.000,
"Back",          3,    0,  1, -1,  0.000,  1.000,  1.000,  1.680,  0.000,  0.000,  -1.000,
"LeftCove0",     4,    0,  1,  0,  0.000,  0.000,  1.000,  0.840,  -0.500,  -0.300,  0.500,
"LeftCove1",     5,    0,  1,  1,  0.000,  0.000,  1.000,  0.840,  -0.500,  0.100,  0.500,
"LeftCove2",     6,    0,  1, -1,  0.000,  0.000,  1.000,  0.840,  -0.500,  0.500,  0.500,
"CenterCove0",   7,    0,  1, -1,  0.000,  0.000,  1.000,  0.840,  -0.400,  0.700,  0.500,
"CenterCove1",   8,    0,  1, -1,  0.000,  0.000,  1.000,  0.840,  -0.200,  0.700,  0.500,
"CenterCove2",   9,    0,  1, -1,  0.000,  0.000,  1.000,  0.840,  0.200,  0.700,  0.500,
"CenterCove3",  10,    0,  1, -1,  0.000,  0.000,  1.000,  0.840,  0.400,  0.700,  0.500,
"RightCove0",   11,    0,  1,  2,  0.000,  0.000,  1.000,  0.840,  0.500,  0.500,  0.500,
"RightCove1",   12,    0,  1, -1,  0.000,  0.000,  1.000,  0.840,  0.500,  0.100,  0.500,
"RightCove2",   13,    0,  1, -1,  0.000,  0.000,  1.000,  0.840,  0.500,  -0.300,  0.500,
};
```

【０１８６】

（注釈文の翻訳）

（注１）

// my_lights.h

//

// ＤｉｒｅｃｔＬｉｇｈｔｓで使用するＣｏｌｏｒ　Ｋｉｎｅｔｉｃｓライトの構成ファイル

//

// このファイルは、ＤｉｒｅｅｔＬｉｇｈｔｓ　ＧＵＩ　Ｓｅｔｕｐ　ｖ１．０によって作成された

（注２）

// 基本構造体をロードする

（注３）

// 全体的なガンマ

（注４）

// どのＤＭＸインターフェースを使用するか？

（注５）

// これは、世界内のすべての実際のライトのリストである

【０１８７】

このファイル例は当社のオフィスから取り出したものであり、オフィスではコンピュー

タの周囲にライトをセットアップし、以下の（モニタの前に座っている人から参照される）ライトを使用している。頭上（ほとんど周囲光）に１つ、頭の各側（左右）に１つずつ、頭の後ろに１つ、目の前にあるモニタの上と左右にそってそれぞれ３つある。

【０１８８】

「ｍｙ＿ｌｉｇｈｔｓ」ファイルの各行は１つのＲｅａｌ＿Ｌｉｇｈｔを表す。各Ｒｅａｌ＿Ｌｉｇｈｔインスタンスは、驚いたことに、１つの現実世界のライトを表す。

モニタの左右の下側のライトは、インジケータ０および２であり、モニタの左側の真ん中のライトはインジケータ１である。

位置の値はメートル単位である。Ｚはモニタの平面の出入りである。Ｘは、モニタの平面内で垂直であり、Ｙは、モニタの平面内で水平である。

【０１８９】

ＭＡＸ＿ＬＩＧＨＴＳは、ＤＭＸユニバース毎に１７０と高い値に設定できる。各ＤＭＸユニバースは、通常、コンピュータとの単一の物理的接続である（例えば、ＣＯＭ１）。ＭＡＸ＿ＬＩＧＨＴＳが大きいほど、ライトの応答は遅くなるが、それはＭＡＸ＿ＬＩＧＨＴＳにより、ＤＭＸに送られるバッファのサイズが決まるからである（ＭＡＸ＿ＬＩＧＨＴＳ＊３）。明らかに、バッファは大きいほど送信に時間がかかる。

ＯＶＥＲＡＬＬ＿ＧＡＭＭＡは、値０～１を取ることができる。この値は、ＤｉｒｅｃｔＬｉｇｈｔに読み込まれるが、実行時に変更することができる。

【図面の簡単な説明】

【０１９０】

【図１】複数の照明システムが配置されている環境を表す図である。

【図２】制御構成要素のグループを使用した複数の光の制御を示す概略図である。

【図３】構成機構およびグラフィック表示機構を使用して照明制御信号を発生する構成要素を示す概略図である。

【図４】アニメーション機構およびライト管理機構から照明制御信号を発生する構成要素を示す概略図である。

【図５】環境内の照明システムに関係するデータのための構成ファイルを示す図である。

【図６】コンピュータのスクリーンを使用した環境の仮想表現を示す図である。

【図７】光を環境の一部分に投射する照明システムを備える環境を表す図である。

【図８】照明システムを介した効果の伝搬を示す概略図である。

【図９】イメージ捕捉装置を使用して環境内の複数の照明システムの位置を決定するステップを示す流れ図である。

【図１０】環境内で照明効果を発生するためにグラフィカル・ユーザ・インターフェースと対話操作するステップを示す流れ図である。

【図１１】ネットワーク送信機により発生されたデータを送信する照明システムを示す概略図である。

【図１２】オブジェクト指向プログラミング手法を使用して照明システム用の制御信号を発生するステップを示す流れ図である。

【図１３】コンピュータのアプリケーションからのデータに基づき現実世界の照明システム用の照明信号を発生するスレッドを実行する流れ図である。

【図１】

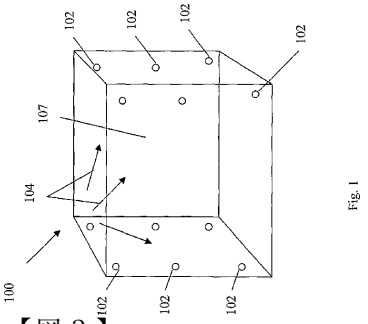
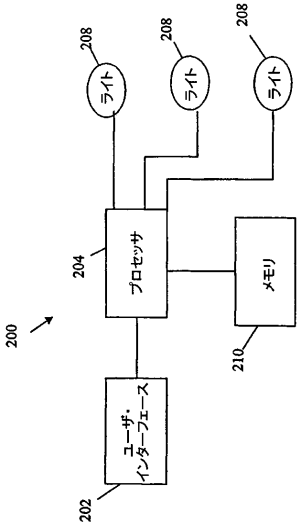
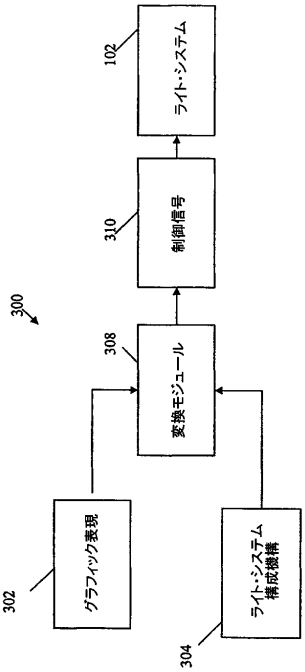


Fig. 1

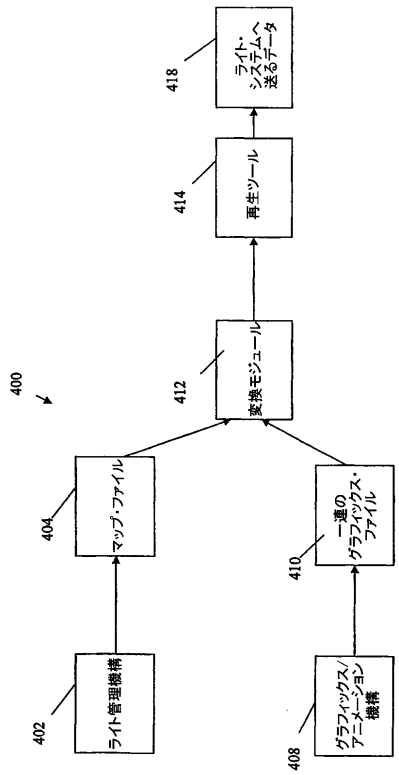
【図２】



【図３】



【図４】



【図５】

500

502 ライト・システム	504 時間	508 位置	510 点灯位置	512 色の範囲	514 光度	518 他のシステム	520 位置
LS001	T1	(1, 3, 7)	Poly001	0-16000	0-100		
LS001	T2	(1, 3, 7)	Poly001	0-16000			
LS001	T3	(1, 3, 7)	Poly 002	0-16000			
LS002	T1	(0,0,0)	Poly 003	0-16000			
LS002	T2	(0,0,0)	Poly004	0-16000			
LS002	T3	(0,0,0)	Poly 005	0-16000			
LS00N	TN						

【図６】

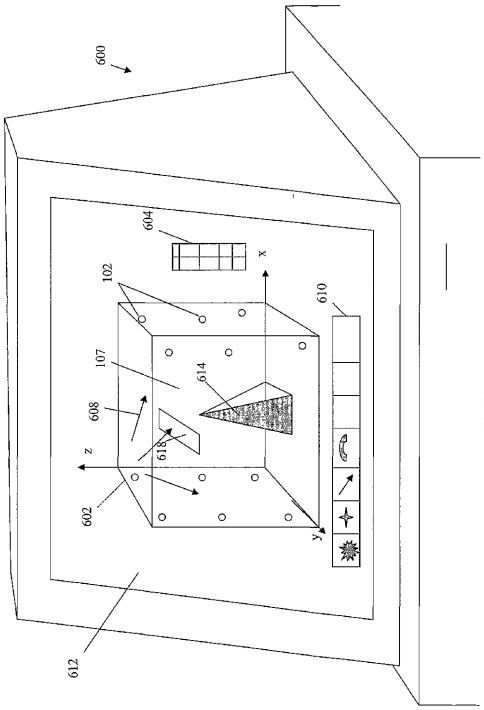


Fig. 6

【図７】

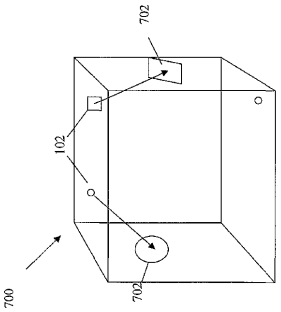
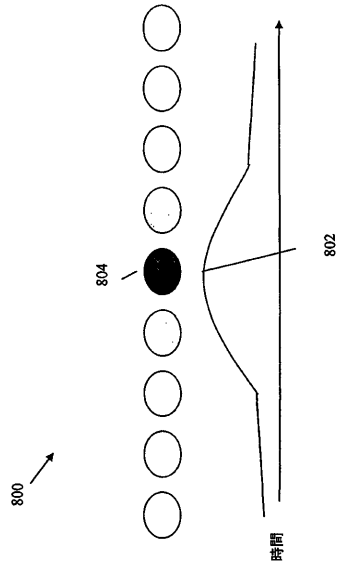
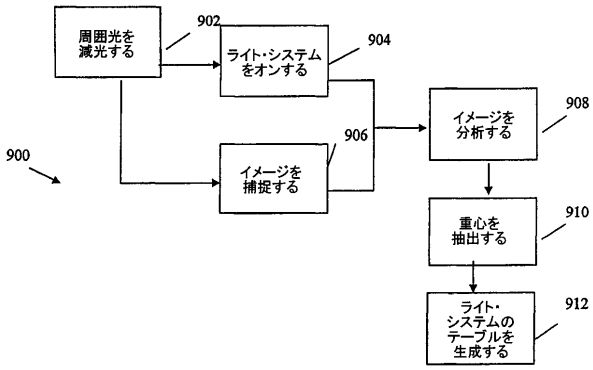


Fig. 7

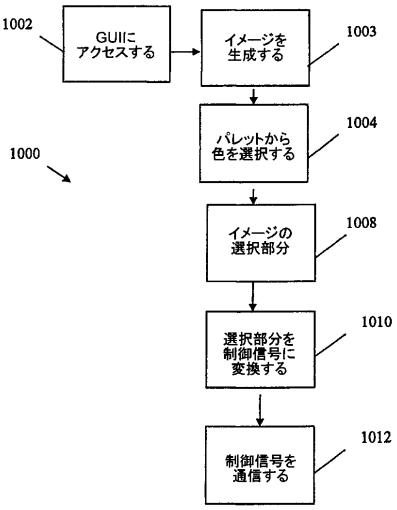
【図８】



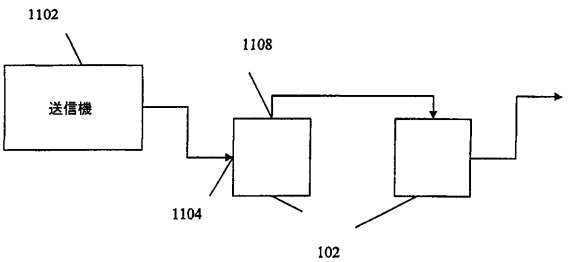
【図９】



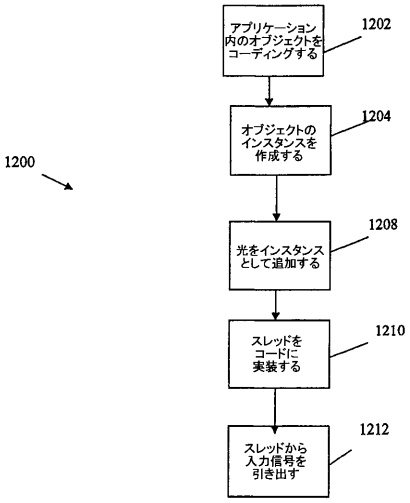
【図１０】



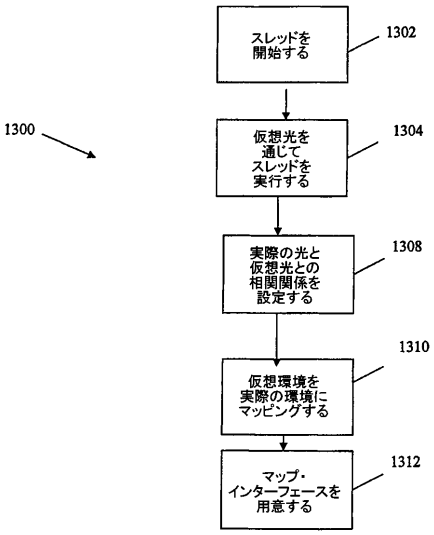
【図１１】



【図１２】


1200
アプリケーション内のオブジェクトをコーディングする
1202
オブジェクトのインスタンスを作成する
1204
光をインスタンスとして追加する
1208
スレッドをコードに実装する
1210
スレッドから入力信号を引き出す
1212

【図１３】


1300
スレッドを開始する
1302
仮想光を通してスレッドを実行する
1304
実際の光と仮想光との相関関係を設定する
1308
仮想環境を実際の環境にマッピングする
1310
マップ・インターフェースを用意する
1312

フロントページの続き

(31)優先権主張番号 09/971,367
(32)優先日 平成13年10月4日(2001.10.4)
(33)優先権主張国 米国(US)
(31)優先権主張番号 60/328,867
(32)優先日 平成13年10月12日(2001.10.12)
(33)優先権主張国 米国(US)
(31)優先権主張番号 60/341,476
(32)優先日 平成13年10月30日(2001.10.30)
(33)優先権主張国 米国(US)

(72)発明者 モルガン，フレドリック・エム
アメリカ合衆国マサチューセッツ州０２１６９，クインジー，バトラー・ロード １５７
(72)発明者 リース，アイハー・エイ
アメリカ合衆国マサチューセッツ州０２１１５，ボストン，ビーコン・ストリート ４７６，アパートメント ６
(72)発明者 チェメル，ブライアン
アメリカ合衆国マサチューセッツ州０１９７０，セーレム，ラファイエット・ストリート ２６０，ナンバー２
(72)発明者 ブラックウェル，マイケル・ケイ
アメリカ合衆国マサチューセッツ州０２１８６，ミルトン，ブルー・ヒルズ・パークウェイ ３５７
(72)発明者 ウォーリック，ジョン
アメリカ合衆国マサチューセッツ州０２１３８，ケンブリッジ，ジョン・エフ・ケネディー・ストリート ２０，アパートメント・ナンバー１

審査官 高橋 学

(56)参考文献 特開２０００－１７３７８３（ＪＰ，Ａ）
特開平０６－０９６６０３（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
H05B 37/00-39/10
F21P 1/00- 7/00