

# モバイル通信環境における 仮想デスクトップ操作応答性能向上技術の検討と評価

山崎浩一<sup>†</sup> 松田正宏<sup>†</sup> 松井一樹<sup>†</sup> 堀尾健一<sup>†</sup>

ユーザが遠隔にある仮想デスクトップ環境を操作する場合，一般的には仮想デスクトップ環境を画面転送方式で転送してユーザの端末に表示する。操作内容や頻度，仮想デスクトップ環境で利用するアプリケーションによっては，大量の画面データを転送する必要があり，ネットワーク帯域幅が狭く，場所や時間によって品質が大きく変動するモバイル通信環境から快適に利用するのは困難であった。本論文では，仮想デスクトップ環境の中でユーザが操作している領域を推定し，推定した領域に含まれる画面データを一定時間内に送信できるサイズに分割送信することで，操作応答性能を向上する方式を提案する。本方式により，モバイル通信環境も含め，帯域幅の狭いネットワーク環境における操作応答性能の向上を実現する。

## Design and Evaluation of the Improvement Technology of Response Time for Mobile Communication Environment

KOICHI YAMASAKI<sup>†</sup> MASAHIRO MATSUDA<sup>†</sup> KAZUKI MATSUI<sup>†</sup>  
KENICHI HORIO<sup>†</sup>

### 1. はじめに

近年，企業や医療，教育機関などさまざまな現場で VDI (Virtual Desktop Infrastructure) の利用が急速に増加している[1]。VDI は端末紛失時の情報漏洩などセキュリティへの対策や自然災害等に対する BCP (Business Continuity Plan / 事業継続計画) の見直し，ユーザのデスクトップ環境の集約によるコスト削減や運用管理の容易性等の観点からニーズが拡大してきている[2][3][4]。さらに，例えば以下のような，これまでにないワークスタイルへの変革に対する効果にも VDI の利用が注目されている[1][5]。

- ・膨大なデータ量となる医療画像やシミュレーション結果のデータを端末にダウンロードせず，サーバ側で生成した画面だけを画面転送方式で閲覧
- ・企業の国内拠点や海外拠点など複数の拠点から 1 台のデスクトップ PC に画面転送方式で接続し，複数拠点で連携したコラボレーションの加速

また，スマートフォンやタブレットなどスマートデバイスの急速な普及，モバイル通信環境の高速化[6]に伴い，外出先等からスマートデバイスを利用し，モバイル通信環境を介して社内のデスクトップ PC や VDI 上の仮想デスクトップ環境に接続して操作する高速モバイルシンクライアント技術[7]のニーズが拡大する

ことが見込まれている。

しかし，モバイル通信環境はネットワーク帯域幅が狭く，通信遅延が大きいだけでなく，時間や場所等によって大きくネットワークの通信品質が悪化する場合があることから，高速モバイルシンクライアント技術における操作応答性能の悪化の要因となっている。

本論文では，特にネットワーク通信品質の低い環境における高速モバイルシンクライアント技術の操作応答性能の改善とその評価を行う。

### 2. モバイル通信環境における課題

高速モバイルシンクライアント技術では LTE や WiMAX 等のモバイル通信環境を介しての利用が多いが，モバイル通信環境のネットワーク品質は，従来の画面転送方式で利用される社内 LAN 環境などのネットワークと比較すると通信品質が低く，頻繁に品質の変動が発生する。LTE 回線を利用した場合の帯域幅は下りで平均 33Mbps という報告[8]があるが，基地局から端末の距離が離れている場合や輻輳が発生している場合などでは，利用可能な帯域幅が 1Mbps 以下となる場合もある。

画面転送方式では，デスクトップ PC や仮想デスクトップ環境の画面表示データ (描画コマンドや画像データ) を取得し，クライアント端末に転送するため，社内 LAN 環境など高品質なネットワーク環境では仮想デスクトップの画面表示データを短時間で転送でき，

<sup>†</sup>1 (株)富士通研究所

操作応答性能も高い。一方、モバイル通信環境や遠隔地からの利用など帯域幅が狭く、遅延の大きいネットワーク環境では、次の画面表示データの転送タイミングまでに画面表示データを転送できず、画面更新が遅れる場合がある。これが続くと、ユーザの操作から仮想デスクトップ環境の画面更新が徐々に遅れていくことになり、ユーザ操作に対する応答性が低下し、操作が困難になる。また、ネットワークの通信品質が変動すると快適に操作できる状態と操作が困難になる状態とが繰り返されることにもなるため、高速モバイルシンクライアント技術の利用に対してユーザが不快に感じる要因となる。

## 2.1 通信品質改善による操作応答性能の向上

このような課題に対し、モバイル通信環境で端末がハンドオーバーした際に TCP 性能の悪化を改善する研究[9]がある。この研究では、ネットワークのレイヤ間制御情報交換機構やハンドオーバー後の有効帯域推定などにより、ハンドオーバー時の通信再開の遅延、転送効率の低下といった問題を改善し、通常の TCP より最大で 4.9 倍の高いスループットを実現している。

高速モバイルシンクライアント技術は、電車等による移動中に利用することもあるが、カフェなど一地点にとどまって利用することも多く、その場合には効果は期待できない。

## 2.2 データ転送量削減による操作応答性能の向上

ネットワークの通信品質に応じてクライアント端末に転送するデータ量を削減し、高速モバイルシンクライアント技術利用時の操作応答性能を向上させる研究[10]がある。この研究では、仮想デスクトップ画面をキャプチャする間隔を予め設定しておき、画面キャプチャからクライアント端末への送信までの時間が予め設定したキャプチャ間隔よりも短ければ仮想デスクトップ画面の更新データをクライアント端末に送信し、長ければ送信しないようにしており、通信品質の悪いネットワーク環境でも操作から更新画面の表示までの遅延時間が延び、操作応答性能が低下するのを抑制している。

この研究では仮想デスクトップ環境のローカルストレージに記録された Web ページを閲覧する評価を行っており、ネットワーク帯域幅 2Mbps では平均遅延時間が約 0.5 秒に短縮しているが、ネットワーク帯域幅が 100Mbps, 10Mbps, 5.5Mbps と狭くなるに伴って平均遅延時間も延びている。これは、ネットワークの通信品質に関わらず、1 フレームあたりのデータ転送量は変わらないためであり、ネットワークの

通信品質が悪化すると平均遅延時間が延び、操作応答性能が低下するためと考えられる。

また、この手法では 1 フレーム分の画面更新データの処理が完了するまで次のフレームが送信されない。そのため、仮想デスクトップの画面サイズや表示されている内容によっては遅延時間が 1 秒以上になることも予想される。その場合、ユーザが操作してからクライアント端末の画面が更新されるまで 1 秒以上待たされることになり、操作応答性能が大きく劣化する。

今後、4K など高解像度の仮想デスクトップ画面を通信品質の低いネットワークを介して利用する場合にはさらに操作応答性能が悪化することが考えられるため、操作応答性能を向上させる技術が重要となる。

## 3. モバイル通信環境における仮想デスクトップ操作応答性能向上技術

高速モバイルシンクライアント技術利用時における操作応答性能の向上には、2.2 で挙げた[10]のように画面更新回数を調整してデータ量を削減するだけでなく、以下の 2 つが重要な要素となると仮定した。

- ① 初動操作から画面更新開始までの初動時間短縮
- ② 操作中における操作追従性の向上

従来の画面転送方式は、以下の 2 つの方式がある。

・ GDI (Graphics Device Interface) などの描画コマンドを転送し、クライアント端末上で描画コマンドを再生することで仮想デスクトップ画面を表示する描画コマンド転送方式

・ 仮想デスクトップ画面を静止画像データや動画像データとして圧縮、転送してクライアント端末上で画像データとして復元することで仮想デスクトップ画面を表示する画像データ転送方式

この 2 方式のうち、描画コマンド転送方式では動画像や CAD など複雑な画面では描画が極端に遅くなる場合がある。

そこで複雑な画面を含め、あらゆる画面での操作応答性能向上を図るため、本論文では画像データ転送方式を対象として検討を行った。

### 3.1 画面更新データの分割送信方法

VNC (Virtual Network Computing) [11]など一般的な画像データ転送方式では、クライアント端末に送信された 1 フレーム分の仮想デスクトップの画面更新データをクライアント端末が一度メモリ上に展開、1 フレ

ーム分すべての画面更新データの展開が完了したタイミングで画面描画処理を行う。この方式では、クライアント端末に表示する仮想デスクトップ画面のちらつきを抑制することができる。しかし、モバイル通信環境など通信品質が悪いネットワーク環境では、[10]の場合と同様に1フレーム分の画面更新データがすべて送信されるまで画面が更新されないため、操作から画面更新までの初動時間が長くなり、操作応答性能は低下する。

通信品質の悪いネットワーク環境で初動時間を短縮する方法として、Frameless Rendering[12]に着目した。Frameless Rendering はフレームの概念を取り除いた映像の表示方法である。この考え方を画像データ転送方式に組み込むとクライアント端末で画面更新データを受信するたびに画面描画処理を行うことになる。この場合、画面のちらつきは発生するが、操作に対する初動時間や画面更新間隔は VNC など従来方式よりも短くなるため、通信品質の悪いネットワーク環境では従来方式と比較して操作応答性能が高くなると考えられる。本論文では、この Frameless Rendering の考え方を高速モバイルシンクライアント技術に取り込み、仮想デスクトップ操作応答性能向上技術として操作から画面更新までの初動時間の短縮を図る。

提案する仮想デスクトップ操作応答性能向上技術では、操作から画面更新までの初動時間や画面更新から画面更新までの更新間隔を更新目安時間とし、ユーザが許容できる更新間隔をユーザ自身が設定できるようにしている。更新目安時間には以下の時間が含まれる。

- ・クライアント端末での操作から仮想デスクトップ側での画面キャプチャまでの時間
- ・画面更新領域抽出時間
- ・画面更新領域の画像データ圧縮時間
- ・クライアント端末への送信時間
- ・圧縮された画面更新データの伸張・描画時間

画面キャプチャまでの時間や画面更新領域抽出時間は仮想デスクトップ画面の画面解像度の変更がない限り変動は小さいため、更新目安時間を長くするとクライアント端末への画面更新データの送信時間や圧縮伸張が長くなり、それに応じて1回の画面キャプチャから画面更新データ送信までのフレーム処理で送信するデータ量(以降、フレーム内送信データ量)も増える。また、更新目安時間を短くすると画面更新データの送信時間や圧縮伸張時間が短くなり、フレーム内送信データ量も減ることになる。そのため、初動時間や更新間隔を短くしようとするとフレーム内送信データ量も減ることになり、1回のフレーム処理ではすべての画面更新データを送信できない。そこで、画面更新データのデータ量がフレーム内送信データ量に収まるよう

に仮想デスクトップ画面を分割して画面更新データのデータ量を減らすことで初動時間や更新間隔が予め設定した更新目安時間と同等程度になるようにする。

一般的な画像データ転送方式の処理は、仮想デスクトップ画面のキャプチャ、前画面からの更新領域の抽出、抽出した更新領域のクライアント端末への送信である。仮想デスクトップ操作応答性能向上技術では、前画面からの更新領域抽出後、仮想デスクトップ画面を分割し、分割した領域の送信優先度を設定する。更新画面データの送信では分割した範囲ごとにその範囲に含まれる画面更新データのみをクライアント端末に送信し、クライアント端末では分割した範囲内の画面更新データをすべて受信すると画面描画処理を行うため、従来方式のように1フレーム分の画面更新データをすべて受信するまで画面描画処理を行う方式と比較して初動時間が短縮することが可能になる。

ここで、フレーム内送信データ量はこの更新目安時間や画面転送方式の各処理における時間、ネットワークの通信品質、主に画面転送方式で利用可能な帯域幅(以降、可用帯域幅)や RTT (Round Trip Time) から以下の式で推測する。

$$D_d = \frac{(T_r - (T_{cap} + T_{cmp} + T_{enc} + T_{dec}) - T_{rtt}) \times B_u}{8} \quad (1)$$

- $D_d$  : フレーム内送信データ量[Byte]
- $T_r$  : 更新目安時間 [sec]
- $T_{cap}$  : 操作から画面キャプチャまでの時間 [sec]
- $T_{cmp}$  : 画面更新領域抽出時間 [sec]
- $T_{enc}$  : 画面更新データの圧縮時間 [sec]
- $T_{dec}$  : 画面更新データの伸張時間 [sec]
- $T_{rtt}$  : ネットワークの RTT [sec]
- $B_u$  : ネットワークの可用帯域幅 [bps]

ここで、画面転送方式における各処理時間は、内部で計測して平均化した値を算出、ネットワークの RTT は画面転送方式の通信とは無関係に決まる値であるため、画面転送方式の接続開始時に算出しておく。可用帯域幅は、[13]の動的画面送信頻度最適化技術における可用帯域幅の推定方法などを利用する。

フレーム内送信データ量が算出できると、仮想デスクトップ画面を分割する。仮想デスクトップ操作応答性能向上技術では分割方向を単純に横方向に設定している。これは画面データが座標(0, 0)から( $X_{max}$ , 0), 座標(0, 1)から( $X_{max}$ , 1),... と横方向に連続データが格納されていることから、仮想デスクトップ画面の分割後に分割した範囲に含まれる画面更新データ抽出等に必要となる計算量を減らし、それにより初動時間や更新間隔が延びることへの影響を小さくするためである。仮



(a) ネットワーク通信品質が高い場合



(b) ネットワーク通信品質が低い場合



(c) 仮想デスクトップ操作応答性能向上技術を適用した場合

図 1：家のモデルを右方向へ移動操作したときの画面表示例

仮想デスクトップ操作応答性能向上技術において分割する際の縦方向のサイズは以下の計算式で算出する。

$$L_d = \frac{D_d}{w \times 3 \times r_c} \quad (2)$$

- $L_d$  : 分割する際の縦方向サイズ[pixel]
- $D_d$  : フレーム内送信データ量[Byte]
- $w$  : 仮想デスクトップ画面の横サイズ[pixel]
- $r_c$  : 画像圧縮による平均圧縮率[%]

ただし、更新目安時間を短く設定しすぎると、式(2)で算出したサイズが非常に小さくなり、クライアント端末で小さい領域しか画面更新されないことになる。その場合、ユーザは画面更新されたと認識できない、あるいは仮想デスクトップ画面の分割数が多くなり、同じ領域における更新間隔が長くなることで操作応答性能が遅いと感じる可能性がある。そのため、最小の分割サイズを予め設定し、算出したサイズが最小の分割サイズ以下の場合には、分割サイズを予め設定した最小の分割サイズに設定するようにした。これにより、算出した分割サイズよりも分割サイズが大きくなることで更新目安時間よりも更新間隔が延びることになるが、ユーザが画面更新を認識できないと操作応答性能が向上したと感じないため、最小の分割サイズを設定することを優先するようにした。分割サイズを決定すると、すべての画面更新データについて仮想デスクトップ画面を分割サイズで分割した領域（以降、分割領域）に振り分け、分割領域ごとに送信優先度を設定する。ここでは、単純に仮想デスクトップ画面の上の分

割領域から下の分割領域に順に送信するように送信優先度を設定した。

また、仮想デスクトップ操作応答性能向上技術では1回の画面キャプチャにつき、1つの分割領域に含まれる画面更新データしかクライアント端末に送信しない。これは、1フレーム分すべての画面更新データを分割領域ごとに送信すると、初動時間は短縮できても操作遅延が拡大する、あるいは、[10]や[13]のようにフレーム調整する場合には、図1のようにマウスなどによる操作に対する画面更新の追従性が悪化するためである。図1の例では、マウスを左から右に移動させて家のモデルを移動させる操作を行った場合のイメージ図である。ネットワークの通信品質が高い場合には(a)のようにマウス操作に追従してオブジェクトが移動するように画面が更新される。一方、通信品質が低い場合には(b)のように操作に対する追従性が悪化することになる。これは、(b)の左から2枚目の時点でキャプチャした画面がクライアント端末で表示されるのが左から4枚目の時点で、左端の画面から左から4枚目での家のモデルの移動量と左から4枚目から右端での家のモデルの移動量が大きく異なるためである。(a)で家のモデルが分割されないのは、ネットワークの通信品質が高く、式(1)の可用帯域幅  $B_u$  が大きいため、フレーム内送信データ量  $D_d$  も大きくなるためである。仮想デスクトップ操作応答性能向上技術では、キャプチャした時点での画面データをクライアント端末に送信するため、(c)のように操作中は家のモデルが大きく分割されることになるが、通信品質が低い場合でも操作に対する

追従性の低下を低減できる。

送信対象となっていない分割領域に含まれる画面更新データは、クライアント端末に送信されないため、分割領域が送信対象となるまで保持しておく必要がある。また、分割領域が送信対象となるまでにさらに画面更新が発生する場合があります、新たに検出された画面更新データと更新領域が重複する場合には、重複した領域を取り除く処理を行う。

これによって操作性向上に重要な要素として仮定した①初動時間の短縮、②操作に対する追従性の向上を実現し、通信品質の悪いネットワーク環境でも高速モバイルシンクライアント技術利用時の操作応答性能を向上することが可能になる。

### 3.2 仮想デスクトップ操作応答性能向上技術の拡張

#### 3.2.1 ユーザの操作領域の抽出

仮想デスクトップの操作では、1つのアプリケーションを画面全体に表示して操作することもあるが、複数のアプリケーションウィンドウが表示されている状態(図2)で操作することが多い。複数のアプリケーションウィンドウが表示されている状態でユーザが操作しているアプリケーションウィンドウ以外の領域で画面更新が頻繁に発生するような場合には、ユーザが操作しているアプリケーションウィンドウ内の画面更新データよりも全体の画面更新データ量が多くなる。通信品質の悪いネットワーク環境では、ユーザが操作しているアプリケーションウィンドウ内の画面更新データがクライアント端末に送信される量が減り、操作応答性能が悪化する。

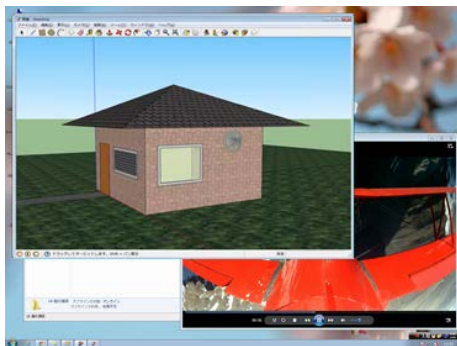


図2. アプリケーションウィンドウの複数表示状態例

そこで、高速モバイルシンクライアント技術利用時にユーザ操作により画面更新が発生している領域(以降、注目領域)を抽出し、抽出した領域を優先して分割・送信することでユーザの操作応答性能の悪化を抑制する。ユーザの操作には、文字の入力、図面の作成、文書や動画の閲覧など多岐にわたるが、ネットワークの通信品質が悪い場合に操作応答性能への影響が大き

い操作として、ここでは図面の閲覧操作を対象に検討、評価を行った。また、本論文では、操作の対象となるモデルはアプリケーションウィンドウの中でも他の領域と比較して多くの色が使われていると想定しており、注目領域の予測・抽出は、次のようにして行う。

まず、仮想デスクトップ画面でアクティブ化されているウィンドウ(以降、アクティブウィンドウ)を検出する。アクティブウィンドウが検出できない場合は、ユーザが特定のアプリケーションを利用していないと判断する。その場合、ユーザはマウスカーソルの付近に注目していると想定し、マウスカーソルの周りを注目領域として設定する。

$$\begin{cases} F_x(x, y) = \left( \left( M_x - \frac{l_f}{2} \right), \left( M_y - \frac{l_f}{2} \right) \right) \\ F_e(x, y) = \left( \left( M_x + \frac{l_f}{2} \right), \left( M_y + \frac{l_f}{2} \right) \right) \end{cases} \quad (3)$$

$F_s(x, y)$  : 注目領域の開始座標

$F_e(x, y)$  : 注目領域の終了座標

$M_x$  : マウスカーソルの x 座標

$M_y$  : マウスカーソルの y 座標

$l_f$  : 注目領域として設定する 1 辺サイズ[pixel]

アクティブウィンドウが検出できた場合は、仮想デスクトップ画面を 8 ピクセル×8 ピクセル、16 ピクセル×16 ピクセルなど縦方向、横方向に  $M_s$  ピクセルごと、格子状に分割し、分割した格子でアクティブウィンドウに含まれる、または一部含まれる格子領域を抽出する。抽出した格子領域について 1 ピクセルごとに色情報を取得、格子領域ごとに使用している色数を計測して予め設定したしきい値以上となる格子領域を、使用色数の多い、複雑な画像領域として設定する。対象となる格子領域すべてについて色数の計測が終了後、すべての複雑な画像領域を取り囲む領域を注目領域として設定し、仮想デスクトップ画面の注目領域以外の領域を非注目領域として設定する。(図3)

#### 3.2.2 注目領域、非注目領域の分割・送信

仮想デスクトップ画面の分割は注目領域と非注目領域で個々に処理する。これは、フレーム内送信データ量の中でより多くの注目領域内の画面更新データをクライアント端末に送信することで、注目領域を検出しない場合と比較して操作応答性能をより向上させるためである。

$$\begin{cases} T_s = T_r - (T_{cap} + T_{cmp} + T_{enc} + T_{dec}) - T_{rtt} \\ D_a = \frac{T_s \times B_u \times P_a}{8} & (\text{注目領域}) \\ D_n = \frac{T_s \times B_u \times (1 - P_a)}{8} & (\text{非注目領域}) \end{cases} \quad (4)$$

$D_a$  : 注目領域のフレーム内送信データ量[Byte]

- $D_n$  : 非注目領域のフレーム内送信データ量[Byte]
- $T_r$  : 更新目安時間 [sec]
- $T_{cap}$  : 操作から画面キャプチャまでの時間 [sec]
- $T_{cmp}$  : 画面更新領域抽出時間 [sec]
- $T_{enc}$  : 画面更新データの圧縮時間 [sec]
- $T_{dec}$  : 画面更新データの伸張時間 [sec]
- $T_{nt}$  : ネットワークの RTT [sec]
- $B_u$  : ネットワークの可用帯域幅 [bps]
- $P_a$  : 注目領域の送信比率

注目領域・非注目領域における送信優先度は、注目領域を検出しない場合と同様、注目領域の一番上の分割領域を一番高く設定し、下の分割領域になるにつれて送信優先度を下げて設定する。注目領域検出時の送信は、まず注目領域の送信対象となっている分割領域内の画面更新データから送信する。送信の際、クライアントに転送したデータ量を積算しておき、送信対象となった分割領域内の画面更新データの総データ量を算出する。注目領域の送信対象となった分割領域内のすべての画面更新データの送信が完了すると、次にフレーム内送信データ量から注目領域の送信対象となった画面更新データ量を減算する。減算した結果が0より大きい場合には非注目領域の送信対象となっている分割領域内の画面更新データを送信し、0以下となる

場合には非注目領域の画面更新データは送信せず、次の画面キャプチャタイミング以降で送信する。(図4)

これにより、ユーザが操作し、注目していると想定される領域の画面更新回数を増やし、注目領域を検出しない場合よりも操作に対する追従性をより向上することが可能となる。

## 4. 評価

画像転送方式を採用した高速モバイルシンククライアント技術に本論文で提案した仮想デスクトップ操作応答性能向上技術を実装し、表1に示す評価環境を利用して評価を行った。今回の評価では一定のネットワークの通信品質を維持するためにネットワークシミュレータを利用して評価を行った。

### 4.1 初動時間の評価

初動時間の評価では、ネットワークの帯域幅を16Mbpsから1Mbpsまで変化(RTTは50ミリ秒固定)させて、ユーザのキー操作からクライアント端末の画面の更新開始までの時間を計測し、提案方式と従来方式の初動時間を評価した。従来方式は[7]の高速モバイルシンククライアント技術を利用した、仮想デスクトップ操作応答性能向上技術の更新目安時間には、操作応

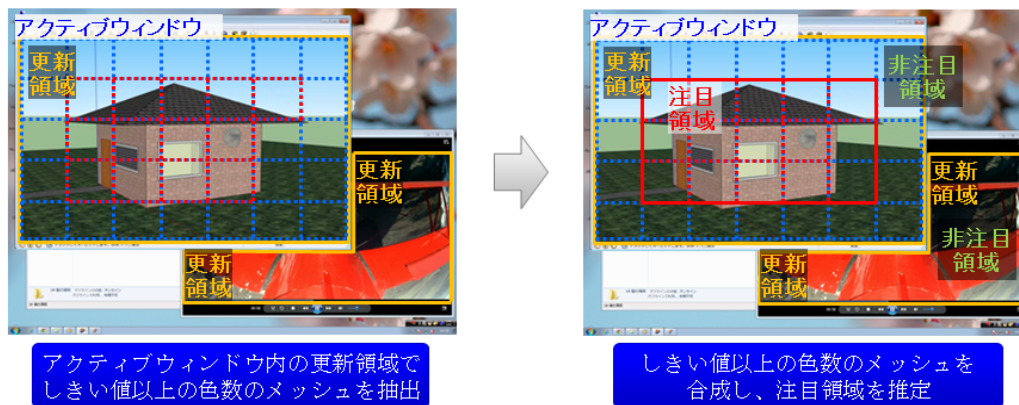


図3：注目領域の推定方法

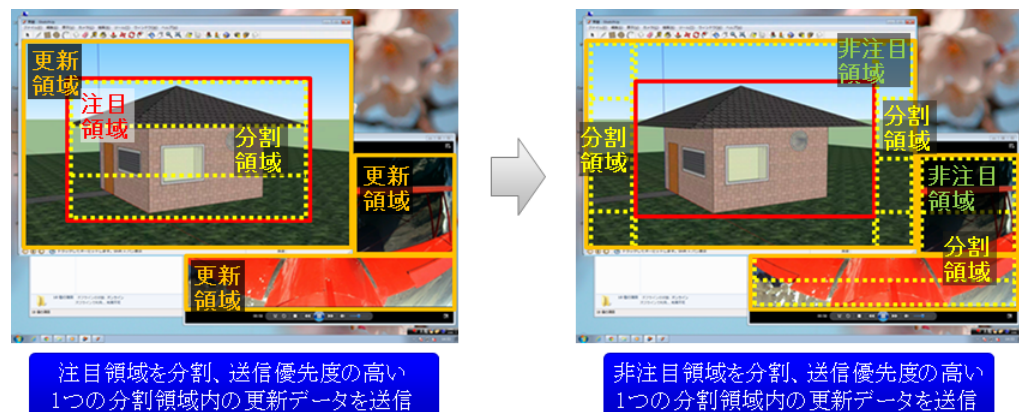


図4：注目領域・非注目領域の分割・送信

表 1：評価環境

	仮想デスクトップ環境	クライアント端末
CPU	Intel Xeon W5590 (3.33GHz) x2	Intel Core i5-3320M (2.60GHz)
メモリ	24GB	8GB
OS	Windows 7 Professional	Windows 7 Professional
画面サイズ	UXGA (1600 x 1200)	UXGA (1600 x 1200)

答性能が快適と感じる上限[16]である 200 ミリ秒を設定した。初動時間はビデオカメラを利用して、クライアント端末でキーを押してからクライアント端末に表示される仮想デスクトップ画面の一部が更新され始めるまでの時間をスローモーション撮影して計測する。

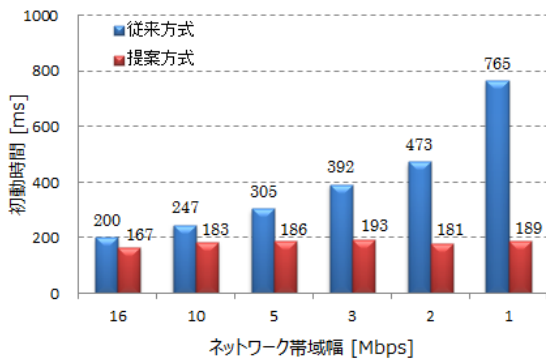


図 5：従来方式と提案方式の初動時間比較結果

計測した結果を図 5 に示す。従来方式はネットワークの帯域幅が狭くなるにつれて、初動時間が徐々に長くなるのに対し、提案方式では、帯域幅に関わらずほぼ一定の初動時間となることが確認できる。また、計測した初動時間は予め設定した更新目安時間と 10 ミリ秒～40 ミリ秒程度の誤差があるが、これは圧縮効率が推測した値よりも高く、圧縮後画面更新データが小さくなり、送信時間が短縮されたこと等が考えられる。

#### 4.2 操作に対する追従性の評価

操作に対する追従性の評価では、従来方式と比較して提案方式のユーザ操作に対する応答性が向上しているかについて主観評価を実施した。主観評価では表 2

に示す 2 つのパターンを用意し、評価方法には、QoE (Quality of Experience)の主観評価法の 1 である PC 法 (Pair Comparison Method)[14][15]を参考にして表 3 に示す評価の尺度を設定し、以下の 2 パターンの評価を実施した。

パターン 1：各ネットワーク帯域幅において従来の方式を基準とした場合の提案方式の操作応答性能

パターン 2：注目領域推定を無効化した提案方式を基準として注目領域推定を有効化した提案方式の操作応答性能

表 3：主観評価での評価の尺度

評点	評定語
-3	操作応答性能が非常に悪化
-2	操作応答性能が悪化
-1	操作応答性能がやや悪化
0	同等の操作応答性能
+1	操作応答性能がやや向上
+2	操作応答性能が向上
+3	操作応答性能が非常に向上

パターン 1 の評価結果を表 4 に、パターン 2 の評価結果を表 5 に示す。評価結果の値は、平均した値となっており、値が大きい方ほど比較対象よりも操作応答性能が良いことを表している。

表 4：パターン 1 の評価結果

帯域幅	5 [Mbps]	3 [Mbps]	2 [Mbps]	1 [Mbps]
評点	+0.6	+1.8	+2.2	+2

表 2：主観評価の評価パターン

	パターン 1	パターン 2
ネットワーク帯域幅	5 [Mbps] ～ 1 [Mbps]	2 [Mbps]
RTT	50 [ms] (往路：25 [ms], 復路：25 [ms])	
評価内容	CAD ソフトを最大化した状態で、3次元オブジェクトの回転・移動操作	CAD ソフトのバックグラウンドで動画再生している状態で、3次元オブジェクトの回転・移動操作。各ウィンドウは通常サイズ(非最大化)

表 5：パターン 2 の評価結果

評点	+1
----	----

パターン 1 の評価では、ネットワーク帯域幅に関わらず提案方式が従来方式と比較すると操作応答性能が向上しているという結果になった。これは、ネットワークの帯域幅にあわせて決定したフレーム内送信データ量によって仮想デスクトップ画面を分割することで従来方式よりも短い間隔でクライアント端末の画面が更新されるために、被験者は操作応答性能が向上していると判断したことが考えられる。また、ネットワークの帯域幅が狭くなるにつれて操作応答性能の向上幅が大きい結果となっている。これは、図 5 の結果からもわかるとおり、従来方式ではネットワークの帯域幅が狭くなるにつれて操作から更新までの時間が伸び、クライアント端末での画面更新回数が減少するのに対し、提案方式では、ネットワークの帯域幅に関わらず、ほぼ同じ時間で画面更新が発生するため、ネットワークの帯域幅が狭くなるにつれて操作応答性能の向上幅が大きく感じたのではないかと考えられる。ただし、帯域幅が 1Mbps の場合に 2Mbps の場合よりも評点が低いのは、分割領域のサイズが最小の分割サイズに設定されたことにより、画面更新データが更新目安時間で送信できるフレーム内送信データ量より大きくなり、送信時間が伸びたことが要因であると考えられる。

パターン 2 の評価では、注目領域を予測・抽出することの有効性を示すために、提案方式で注目領域を予測・抽出しない場合との比較評価を実施した。この評価では、注目領域を予測・抽出しない場合と比較して操作応答性能が向上したと判断・評価されていることが確認できた。これは、注目領域を予測・抽出しない場合、オブジェクトの操作による更新以外にも動画再生の更新を同時に送信することになるため、操作している領域について一度に送信できるデータ量が小さくなるためであり、今回の提案方式の操作応答性能が向上したという結果になったのは、注目領域を予測・抽出することで一度に送信できるデータ量が増えたためであると考えられる。

## 5. まとめ

本論文で提案した仮想デスクトップ操作応答性能向上技術を適用した高速モバイルシンクライアント技術を開発し、様々な帯域幅のネットワーク環境において操作応答性能が改善することを確認した。

今回、仮想デスクトップ環境の画面解像度は UXGA(1600x1200)サイズで評価を実施したが、4K/8K

など更なる高解像度画面になると今回提案した方式による分割数が非常に多くなり、4.2 で 1Mbps の評価結果のように操作応答性能の向上が小さくなる、または向上がないと感じることも考えられる。そのため、今後もモバイル通信環境での実用性調査を継続し、4K/8K など仮想デスクトップ画面の高解像度環境で利用する場合やさらにネットワーク帯域が狭く、RTT の大きい海外から接続して利用する場合など、さらなる操作性改善に向けた課題抽出と対策を実施する。

## 参考文献

- 1) IDC Japan：国内クライアント仮想化市場予測を発表、入手先 <<http://www.idcjapan.co.jp/Press/Current/20131211Apr.html>> (2013.12.11)
- 2) 梅澤 克之, 手塚 悟：スマートホンをセキュアデバイスとして用いるリモート接続システムの開発と評価, 電子情報通信学会論文誌 B, Vol. J94-B, No.4, pp. 530-538 (2011)
- 3) 樋地 正浩：東日本大震災における情報通信技術の利用と課題, 電子情報通信学会論文誌 D, Vol.J95-D, No.5, pp.1070-1080 (2012)
- 4) Kwon, E., Yoon, C. and Ryu, W.: A novel device sharing scheme for thin client, Computing and Convergence Technology (ICCCT), 2012 7th International Conference, pp.481-484 (2012)
- 5) Humar, I., Bester, J., Tomazic, S.: Characterizing Graphical Desktop Sharing System's Workload in Collaborative Virtual Environment, Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE, pp.1-5 (2009)
- 6) Tang, W., Song, B., Kim, S.M., Dung, T.N., Huh, N.E.: Hybrid remote display protocol for mobile thin client computing, Computer Science and Automation Engineering (CSAE), 2012 IEEE International Conference (2012)
- 7) Imai, T., Horio, K., Ohno, T., Matsui, K.: An adaptive desktop transfer protocol for mobile Thin Client, GLOBECOM Workshops, 2010 IEEE, 10.1109/GLOCOMW.2010.5700112, pp.1136-1140 (2010)
- 8) Elnashar, A. and El-Saidny, A.M.: Looking at LTE in Practice, IEEE VEHICULAR TECHNOLOGY MAGAZINE, pp.81-92 (2013)
- 9) 韓 閔燮, 寺岡 文男：クロスレイヤーアーキテクチャに基づくハンドオーバー時の TCP のふくそう制御手法, 電子情報通信学会論文誌 B, Vol. J91-B, No.10, pp.1171-1181 (2008)
- 10) Shen, H., Lu, Y., Wu, F. and Li, S.: A high-performance remote computing platform, Pervasive Computing and Communications, 2009. PerCom 2009. IEEE International Conference, pp.1-6 (2009)
- 11) Richardson, T., Stafford-Fraser, Q., Wood, K.R., Hopper, A.;; "Virtual network computing," Internet Computing, IEEE, vol.2, no.1, pp. 33-38 (1998)
- 12) Bishop, G., Fuchs, H., McMillan, L., and Scher Zagier, J.L.: "Frameless Rendering: Double Buffering Considered Harmful", Proc. ACM SIGGRAPH, pp. 175-176 (1994)
- 13) 堀尾 健一, 松井 一樹：モバイルシンクライアント向け操作応答性能向上技術の検討および評価, マルチメディア, 分散, 協調とモバイル(DICOMO2012)シンポジウム, pp.1847-1854 (2012)
- 14) ITU-T: Recommendation P.800 "Method for subjective determination of transmission quality," Aug. 1996
- 15) 岡本 淳, 林 孝典：映像メディアの品質評価技術の最新動向, 電子情報通信学会, 基礎・境界ソサイエティ, Fundamentals Review Vol.6, No.4, pp.276-284 (2013)
- 16) Ramamurthy, S., Helal, A.: Optimising thin clients for wireless computing via localization of keyboard activity, Performance, Computing, and Communications, 2001. IEEE International Conference, 10.1109/IPCCC.2001.918659, pp.249-252 (2001)